ESCOLA SUPERIOR ABERTA DO BRASIL – ESAB
PÓS-GRADUAÇÃO LATO SENSU DE ENGENHARIA DE SISTEMAS

GUILHERME FERREIRA RIBEIRO

USABILIDADE DE INTERFACES COM O USUÁRIO DE SOFTWARES DE EDIÇÃO E EDITORAÇÃO DE PARTITURAS

VILA VELHA - ES
2010

GUILHERME FERREIRA RIBEIRO

USABILIDADE DE INTERFACES COM O USUÁRIO DE SOFTWARES DE EDIÇÃO E EDITORAÇÃO DE PARTITURAS

Monografia apresentada à ESAB – Escola Superior Aberta do Brasil, sob orientação da Professora Maria Ionara Barbosa de Andrade Gonçalves

VILA VELHA - ES
2010

GUILHERME FERREIRA RIBEIRO

USABILIDADE DE INTERFACES COM O USUÁRIO DE SOFTWARES DE EDIÇÃO E EDITORAÇÃO DE PARTITURAS

Aprovada em de de 2010

_____

_____

_____

VILA VELHA - ES
2010

À Rosali e Rafaela.

AGRADECIMENTOS
À minha família. A todos que colaboraram com a pesquisa online, especialmente aos amigos do fórum violão.org. À Maria de Fatima pela ajuda com a revisão do texto.

“Tudo está escrito numa partitura, exceto o essencial.”
(Gustav Mahler)

RESUMO

Este trabalho descreve a aplicação de técnicas de avaliação de usabilidade nas interfaces dos softwares mais utilizados para edição e impressão de partituras. Foi realizada uma análise heurística de cada um dos softwares selecionados e foram coletadas opiniões de usuários através de um questionário online. Também foram coletadas informações em grupos de discussão na internet e entrevistas informais com usuários. Segue uma discussão sobre os problemas de usabilidade potenciais encontrados e suas consequências.

SUMÁRIO

RESUMO..... 7
INTRODUÇÃO. 10
EXPOSIÇÃO DO ASSUNTO 10
PROBLEMA DE PESQUISA..... 10
JUSTIFICATIVA PARA A ESCOLHA DO TEMA 10
OBJETIVO GERAL.....11
OBJETIVOS ESPECÍFICOS 11
DELIMITAÇÃO DO TRABALHO 12
METODOLOGIA DE PESQUISA 12
CAPÍTULO I – FUNDAMENTAÇÃO TEÓRICA..... 13
I.1- NOTAÇÃO MUSICAL 13
I.1.1 - Notação Musical – História 13
I.1.2 - Desenho e Impressão de Partituras – História..... 15
I.1.3 - Notação Musical – Elementos Básicos..... 19
I.2 - USABILIDADE..... 23
I.2.1 - Atributos de Usabilidade..... 23
I.2.1.1 - “Aprendabilidade” 23
I.2.1.2 - Eficiência..... 24
I.2.1.3 - “Memorabilidade”..... 25
I.2.1.4 - Erros..... 26
I.2.1.5 - Satisfação..... 26
I.2.2 - Categorias de Usuários..... 26
I.2.3 - Heurísticas de Usabilidade 30
I.2.3.1- Heurísticas de Nielsen..... 30
I.2.3.2 - Critérios de Scapin/Bastien 33
I.2.4 - Métodos de Avaliação de Usabilidade 34

I.2.4.1 - Questionários e Entrevistas 35
I.2.4.2 - Avaliação Heurística 36
I.2.4.3 - Testes de Usabilidade 37
I.3 - DETALHAMENTO DO ESCOPO DAS AVALIAÇÔES 38
CAPÍTULO II – RESULTADOS 40
II.1 - AVALIAÇÃO HEURÍSTICA 40
II.1.1 - Encore 40
II.1.2 - Finale 46
II.1.3 - Sibelius 52
II.2 - QUESTIONÁRIO 58
III - CONCLUSÃO 73
ANEXO I - QUESTIONÁRIO 75
ANEXO II – GRÁFICOS (REPOSTAS DO QUESTIONÁRIO) 80
ANEXO III – PERGUNTA 19 (QUESTIONÁRIO) 86
PONTOS POSITIVOS 86
Encore 86
Finale 86
Lilypond 87
Sibelius 87
TuxGuitar 87
PONTOS NEGATIVOS 87
Encore 87
Finale 88
Lilypond 88
ANEXO IV – PERGUNTA 20 (QUESTIONÁRIO) 89
REFERÊNCIAS BIBLIOGRÁFICAS 91

INTRODUÇÃO ENGENHARIA DE SOFTWARE USABILIDADE EDITORAÇÃO DE PARTITURAS**.**

EXPOSIÇÃO DO ASSUNTO

Este trabalho busca aplicar as práticas da Engenharia de Usabilidade na avaliação das interfaces com o usuário dos softwares de edição e editoração de partituras mais usados.

PROBLEMA DE PESQUISA

Qual o grau de usabilidade das interfaces com o usuário dos softwares de edição e editoração de partituras mais usados?

JUSTIFICATIVA PARA A ESCOLHA DO TEMA

A usabilidade é um atributo ao qual se tem dado cada vez mais importância e ao qual se tem dedicado cada vez mais recursos no processo de desenvolvimento de softwares. A otimização da usabilidade do software visa a promover um diálogo do usuário com o sistema centrado no usuário e nas tarefas para as quais ele usa o sistema, e não em limitações impostas por plataformas de hardware, linguagens de programação ou metodologias de desenvolvimento.

A editoração de partituras é uma aplicação que apresenta vários desafios em relação a este atributo. A notação musical é uma linguagem gráfica complexa, com requisitos técnicos e estéticos rigorosos para os documentos produzidos; os usuários são principalmente artistas (músicos, compositores, arranjadores), profissionais ou amadores, e a frustração com uma interface com problemas de usabilidade pode causar, além dos problemas comuns a outras áreas, até mesmo um bloqueio da

inspiração, fundamental em uma atividade criativa.

Uma avaliação das interfaces dos softwares mais utilizados nesta aplicação, além de uma oportunidade de empregar as práticas da Engenharia de Usabilidade, pode ainda fornecer subsídios para a escolha de um software específico, servir de ponto de partida para o desenvolvimento de novos sistemas, ou levantar possibilidades para o aperfeiçoamento dos softwares avaliados.

OBJETIVO GERAL

Avaliar a usabilidade das interfaces com o usuário dos softwares de edição e editoração de partituras mais usados através da estimação de atributos de usabilidade.

OBJETIVOS ESPECÍFICOS

Descrever a interação dos usuários com os softwares mais usados na realização de algumas tarefas comuns ou importantes na edição e editoração de partituras.

Definir atributos de usabilidade aplicáveis e técnicas de avaliação para estes atributos.

Utilizar estas técnicas para realizar uma avaliação da interface destes softwares de acordo com os atributos de usabilidade escolhidos, levantando problemas de usabilidade potenciais.

Fazer uma comparação entre os softwares testados de acordo com os resultados obtidos.

Avaliar o grau de usabilidade desta categoria de softwares baseado nesta amostra.

DELIMITAÇÃO DO TRABALHO

Foram objeto das avaliações apenas softwares com interfaces gráficas interativas, sendo que softwares com paradigmas de interface de gerações anteriores foram citados quando relevante. O estudo se focou nos softwares mais populares e com maior participação no mercado. As técnicas de avaliação de usabilidade foram escolhidas de acordo com a viabilidade de tempo e recursos, buscando-se as que são passíveis de aplicação sem deslocamento de usuários até locais de teste e sem necessidade de laboratórios especiais. Nas técnicas onde é necessário mais de um avaliador especializado em usabilidade, a aplicação foi simplificada apresentando-se os resultados obtidos por apenas um pesquisador.

METODOLOGIA DE PESQUISA

Foi realizada uma pesquisa de campo, com a coleta de dados de usuários através de um questionário on-line. Os dados coletados foram tratados como indicadores de problemas de usabilidade potenciais (ou de pontos fortes da usabilidade do ponto do vista do usuário) e não como objeto de uma análise estatística tradicional. Utilizou-se também o método de Avaliação Heurística (NIELSEN, 1993) para a procura de problemas potenciais de usabilidade.

CAPÍTULO I – FUNDAMENTAÇÃO TEÓRICA

I.1- NOTAÇÃO MUSICAL

O objetivo desta seção é apresentar, de forma concisa, a notação musical e as tecnologias usadas para criação de partituras, suas origens, sua evolução e seus elementos principais. A intenção é familiarizar os leitores da área de sistemas com os termos do domínio da aplicação, a música, e mostrar o contexto em que se encontram os softwares que serão avaliados neste trabalho.

I.1.1 - Notação Musical – História

Notação musical é uma linguagem gráfica usada para representar obras musicais. É uma forma de registro do conteúdo musical, assim como um conjunto de instruções para que músicos possam interpretar este registro. A notação musical utilizada no mundo ocidental, que é o tipo de notação considerado neste trabalho, usa um grande número de símbolos e o posicionamento destes símbolos para representar a informação musical. Comparada com outras linguagens, é uma linguagem complexa, porém lógica e bem organizada. É eficiente, sendo capaz de representar praticamente toda a produção musical (ocidental) desde o século XVII até a primeira metade do século XX, e, apesar de não existir um padrão “oficial”, permitindo a execução por músicos independentemente de diferenças geográficas, culturais ou temporais (até mesmo à primeira vista, dependendo da habilidade técnica dos músicos e do grau de dificuldade da obra musical) (LASSFOLK, 2004).

A notação musical utilizada no mundo ocidental tem sua origem atribuída ao monge Guido D'Arezzo no século XI. Seu objetivo era registrar os cantos utilizados nas missas, até então normalmente preservados apenas por memorização e transmitidos oralmente. O sistema de D'Arezzo utilizava pequenos quadrados representando as notas. A sequência temporal das notas era indicada pela disposição horizontal dos símbolos (da esquerda para a direita), enquanto a posição

vertical dos símbolos indicava a altura relativa de cada nota. Posteriormente, foram acrescentados um conjunto de linhas horizontais que facilitava discriminar qual a posição vertical (e portanto a altura relativa) e uma letra chamada “clave” que determinava a altura absoluta representada por cada linha (e pelos espaços entre elas).

Esta notação foi sendo aperfeiçoada para atender às necessidades dos músicos e compositores, principalmente a da representação precisa do ritmo (a duração relativa de cada nota). Foram criados símbolos que são anexados a cada nota, como um adorno. Quanto mais rica em adornos, menor a duração da nota, segundo uma progressão em que, em geral, cada adorno acrescentado reduz a duração à metade. Este sistema é atribuído a Franco d'Cologne no século XIII e aperfeiçoado e expandido por Philippe d'Vitry no século XIV. Para representar a pulsação rítmica, no século XV, foram acrescidos símbolos de frações e, no século XVII, apareceram barras verticais para separação de séries regulares de tempos. Com a adoção do sistema igualmente temperado (divisão da oitava em doze partes iguais), no período barroco, foi padronizada uma forma de representação da tonalidade: a armadura de clave (“PLAY IT AGAIN! PIANO STUDIO”) (BLOOD).

A notação como é utilizada hoje já estava praticamente consolidada no século XVII. No século XX, entretanto, compositores de vanguarda começaram a introduzir elementos em suas composições que muitas vezes sequer eram considerados como “musicais” no passado. A notação convencional é limitada para registrar este tipo de composição, e muitos compositores passaram a criar seus próprios símbolos e regras. As partituras destas composições costumam incluir um conjunto de instruções para interpretar estes novos símbolos, chamado “bula”. Não é infrequente a bula chegar a possuir mais páginas do que a composição em si. A flexibilidade para permitir a criação deste tipo de partitura é uma característica desejável para ferramentas computacionais de edição de partituras, mas, neste trabalho, o foco será restrito à notação convencional.

I.1.2 - Desenho e Impressão de Partituras – História

Desde os primeiros manuscritos até o uso de computadores, acompanhando a evolução da imprensa, várias tecnologias foram utilizadas (e muitas continuam em uso) para o desenho e impressão de partituras.

Um dos primeiros métodos, surgido no século XV, é o da xilogravura. Um bloco de madeira era entalhado em relevo com um negativo espelhado da partitura. Era aplicada tinta sobre este relevo e o bloco era usado para estampar o papel ou pergaminho.

Com a invenção da imprensa e da tipografia em 1450 (Gutenberg), as partituras também passaram a ser impressas utilizando tipos móveis. Pequenos blocos de metal, cada um com um determinado símbolo em relevo, eram arrumados como em um quebra-cabeças de modo a formar a partitura completa que seria impressa com o auxílio de uma prensa.

Como os tipos móveis não permitiam registrar todas as nuances da escrita manual, foram criadas novas técnicas. O processo de gravura usa estiletes, punções e outras ferramentas para gravar uma imagem da partitura numa placa de metal, que então é usada para imprimir um negativo. O resultado gráfico é excelente, e era tão popular que em inglês o termo tradicionalmente usado para editoração de partituras é “music engraving”. Continuou sendo usado, mesmo após a disseminação do uso da edição por computadores, até o início do século XXI.

A litografia, inventada no final do século XVIII, consistia em desenhar a partitura sobre uma placa de pedra usando uma tinta à base de óleo e depois usar um ácido para corroer as áreas da pedra não cobertas pela tinta. Este bloco era então usado para imprimir as partituras. Era uma alternativa mais econômica para a publicação de partituras, e posteriormente evoluiu para técnicas como a cromolitografia e a fotolitografia.

O desenho das partituras, para impressão por alguma das técnicas citadas acima, era principalmente feito à mão, utilizando ferramentas de desenho como penas,

gabaritos e curvas francesas, até quando houve a disseminação do uso de computadores para este fim. No início do século XX, foram inventadas máquinas de escrever especializadas para música, mas podem ser consideradas apenas uma curiosidade.

Nos anos 1950 e 1960, os avanços tecnológicos tornaram possível a utilização de computadores para criação de partituras. Os primeiros programas criados nos anos 1960 foram o Plaine and Easie Code e o DARMS (Digital Alternate Representation of Musical Scores). Posteriormente, foi desenvolvida uma máquina chamada ILLIAC que lia uma tira de papel perfurado contendo uma codificação da partitura e produzia uma impressão formatada.

Em 1977, foi criado o Musicomp (fig. 1), um computador dedicado à criação de partituras. Ele possuía dois teclados, um para determinar a altura das notas e outro contendo símbolos e letras. A música era mostrada numa tela e até 30 páginas podiam ser armazenadas numa fita cassete.

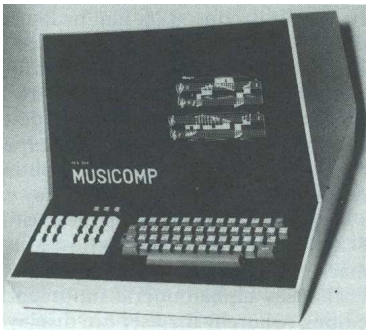

Figura 1 – Musicomp.
Fonte: (Reyna, 2009)

Com o aparecimento dos computadores pessoais, vários programas surgiram, como SMUT, SCORE, MUSTRAN, MEG, e o Oxford Music Processor. O SCORE, desenvolvido para MS DOS, recebe um arquivo com uma série de comandos em uma linguagem específica e cria uma partitura gráfica pronta para a impressão.

Devido a sua qualidade excelente, flexibilidade e alto grau de controle do resultado final por parte do usuário, ainda hoje é o software mais utilizado profissionalmente na área de publicação de partituras, mesmo sendo baseado no processamento em batch e concorrendo com aplicativos que dispõem de interfaces gráficas interativas com manipulação direta e visualização WYSIWYG instantânea.

Nos anos 1980, o aparecimento do MIDI (Musical Instrument Digital Interface), um protocolo de comunicação serial entre instrumentos musicais eletrônicos, permitiu o uso de instrumentos musicais, principalmente teclados eletrônicos, como dispositivo de entrada para a edição de partituras por computador, além de permitir a audição do resultado através de sintetizadores de som. Um dos primeiros programas a explorar esta capacidade foi o Mockinbird.

Com o início do uso de interfaces gráficas nos computadores pessoais, programas como o Professional Composer passaram a permitir o uso do mouse e paletas de símbolos para a criação de partituras (REYNA, 2009).

Em 1990, a Passport Designs, uma das pioneiras na criação do MIDI, lançou o programa Encore, que possuía uma interface que até hoje é considerada uma das mais fáceis de aprender e usar, e foi muito popular durante o início dos anos 1990. Praticamente todos os elementos da partitura permitiam manipulação direta na tela. Entretanto, por problemas como o aparecimento de concorrentes e uma série de incompatibilidades (com novas versões do MS-Windows e vários drivers de impressoras, por exemplo), o Encore acabou perdendo mercado. Em 1998 a Passport Designs foi vendida para a GVox Inc. e, enquanto os concorrentes lançam versões novas a cada ano com cada vez mais aperfeiçoamentos e novas funcionalidades, somente em 2008 (após um lapso de dez anos), uma nova versão do Encore foi lançada. A versão de demonstração instalada para a realização deste trabalho se mostrou extremamente instável, tornando quase impraticável seu uso. Uma consulta ao fórum de suporte da GVOX na internet revelou que realmente a nova versão apresenta problemas e não foi um caso isolado. Como não é possível adquirir de maneira legal uma versão mais antiga, foi cogitado deixá-lo de fora dos testes. Convém notar que apesar destes problemas o Encore ainda possui uma base grande de usuários fiéis, talvez devido a uma real qualidade de sua interface. Isto

pode ser comprovado através do número de sites que oferecem partituras no formato Encore para download que pode ser encontrado numa busca na internet (THEORETICALLY CORRECT, 2010).

Atualmente existe uma grande variedade de softwares disponíveis, mas o mercado de softwares interativos de edição de partituras é dominado por dois programas: Sibelius e Finale. Pode-se notar inclusive certa polarização entre os usuários dos dois softwares.

A primeira versão do Finale foi lançada em 1988 pela Coda Music Software, que depois foi vendida para a Net4Music e depois se tornou a MakeMusic. Durante muito tempo, ela dominou o mercado praticamente sozinha. (BUSINESS WIRE, 2002)

O desenvolvimento do Sibelius começou em 1986, e em 1993, foi lançada a primeira versão ao público. Ele rodava em computadores Acorn. Apesar desta plataforma não ser muito difundida, ela era bastante usada em escolas britânicas, e devido a isto o Sibelius fez bastante sucesso na Inglaterra (BIELAWA, 1997). Em 1998, foram lançadas as primeiras versões para MacIntosh e PC. Nesta época, o domínio do Finale era quase total, mas a participação do Sibelius aumentou rapidamente (CULLEN, 2006). Em uma entrevista em 2007, um executivo da Sibelius Software afirmava que eles detinham sozinhos 50% do mercado americano. (BROWN, 2007)

Os usuários do Sibelius apontam uma melhor usabilidade da interface como a razão da sua preferência. Os usuários do Finale, por outro lado, dizem que o aprendizado inicial no Sibelius seria mais fácil, mas o Finale permitiria uma maior eficiência e qualidade do resultado final para usuários experientes. Vale a pena notar que tanto o Sibelius quanto o Finale são na verdade famílias de produtos, sendo disponíveis diversas versões voltadas para diferentes grupos de usuários, com preços e funcionalidades diferenciados.

Este histórico mostra que este setor sempre procurou acompanhar e tirar partido da evolução tecnológica tanto de hardware quanto de interfaces, e que a usabilidade é um fator muito influente para o sucesso de um software neste mercado. Entretanto, como já citamos no caso do Score, usuários experientes podem preferir abrir mão de

uma interface gráfica e trabalhar diretamente com a compilação de sequências de códigos em um processamento em batch, se a melhoria na qualidade do produto final for compensadora. O Lilypond é um software deste tipo, inspirado no editor de textos T_EX, e vem ganhando muita popularidade recentemente. Ele é gratuito, multi plataforma e de código aberto, e produz um resultado final de alta qualidade (LILYPOND, 2009).

Existem ainda softwares especializados em determinados tipos de notação, por exemplo, o Guitar Pro, voltado para a criação de partituras e tablaturas para violão, guitarra e outros instrumentos de cordas dedilhadas. Por um lado, este tipo de software procura priorizar algumas tarefas específicas que podem exigir mais esforço em um programa de uso mais geral, porém em detrimento de outras funcionalidades. Muitos tipos de software, como os DAW (Digital Áudio Workstation), por exemplo Cubase e Cakewalk Sonar, implementam alguma capacidade de edição de notação musical, embora não seja este seu objetivo principal. Além de normalmente possuírem menos recursos voltados à edição de partituras, a qualidade das partituras produzidas tende a ser menor em comparação aos softwares dedicados. Estes dois tipos de software, portanto, não fazem parte do escopo deste trabalho.

I.1.3 - Notação Musical – Elementos Básicos

Nesta seção, será apresentado um resumo bastante sucinto dos elementos mais importantes da notação musical, baseado em (MOODS PROJECT, 2009) e (MED, 1980).

As *notas* são símbolos gráficos que representam a *altura* e a *duração* de um som musical. A altura é determinada pela sua posição em uma *pauta* composta por *linhas* e *espaços*. Um símbolo chamado *clave* assinala na pauta uma referência fixa de altura. Uma combinação de pautas é chamada *sistema*. (fig. 2)
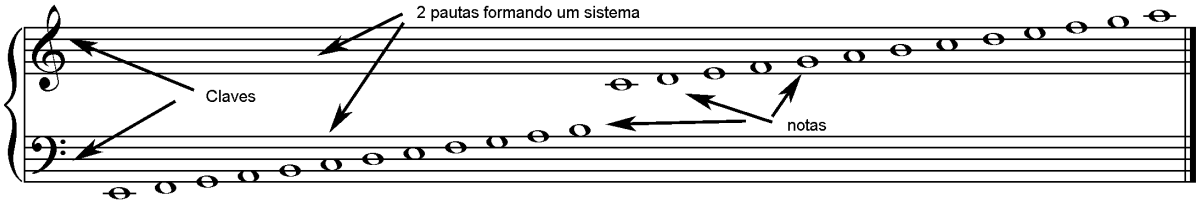
Figura 2 – Sistema, pauta, claves de sol e de fá, notas.
Fonte: ilustração gerada pelo autor usando o software Lilypond.

O símbolo mais simples para uma nota é um círculo vazio, a *cabeça* da nota. Ao acrescentar-se a este círculo mais elementos como a *haste*, o preenchimento da cabeça e um ou mais *colchetes*, modifica-se a duração da nota (cada elemento acrescentado reduz a duração à metade). Cada símbolo de duração de nota tem um símbolo de pausa equivalente para representar um silêncio com a mesma duração (fig. 3). Os colchetes podem ser substituídos por uma *barra de ligação* criando um agrupamento visual de notas (fig. 4). Outros elementos modificadores da duração são o *ponto de aumento* (cada ponto aumenta a duração em 50%) e a *ligadura*, que permite combinar a duração de várias notas representando um único som. A *quiáltera* permite modificar a duração de um grupo de notas de modo que ele tenha a mesma duração de um grupo com um número de notas diferente.

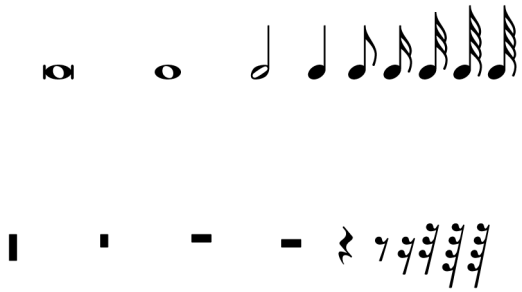

Figura 3 – Notas e pausas em ordem decrescente de duração.
Fonte: ilustração gerada pelo autor usando o software Lilypond.

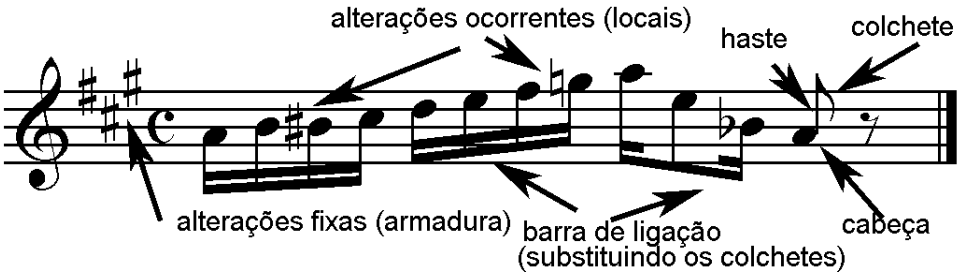

Figura 4 – Alterações, partes da nota.
Fonte: ilustração gerada pelo autor usando o software Lilypond.

As *alterações* (*sustenido*, *bemol*, *dobrado sustenido*, *dobrado bemol*, *bequadro*) permitem modificar a altura de uma nota de modo que com apenas sete nomes de notas e mais as alterações, podemos representar as doze divisões da oitava que se usam na música ocidental. Uma alteração é *fixa* ou *tonal*, quando é parte de um conjunto de alterações indicado junto à clave, chamado *armadura de tom*, e vale para toda a partitura ou até ser substituída por uma nova assinatura de tonalidade. Outro tipo de alteração é a *ocorrente*, que só tem validade no contexto local de um compasso (fig. 4).

Um *compasso* é uma divisão da música em séries regulares de *tempos* (pulsações), separados na partitura por *barras* verticais. Uma *fórmula de compasso* é um símbolo de fração no início de um trecho musical no qual o numerador identifica o número de pulsações por compasso e o denominador identifica qual símbolo de duração de nota equivalerá a uma pulsação. Algumas fórmulas mais usadas possuem símbolos especiais (fig. 5).

A notação musical permite, além da representação de melodias como uma sequência de notas, a representação de *acordes* (fig. 5), que são notas tocadas simultaneamente, ou mesmo a representação de várias melodias (*vozes*) concorrentes na mesma pauta (*polifonia*), usando recursos como a direção das hastes ou o tamanho da cabeça para discriminar cada voz.

Além destes elementos básicos, existem outros símbolos que permitem, por exemplo, indicar e modificar o *andamento*, a *articulação*, a *dinâmica* e a *expressão*.

Estes símbolos podem ser textuais (geralmente em italiano) ou gráficos (fig. 5).

Para uma representação mais eficiente, existem também símbolos de *abreviaturas* como repetição de trechos ou notas e *ornamentos* (embelezamentos melódicos comuns) (fig. 5).

Existem ainda elementos específicos para o instrumento a que se destina a partitura, como digitações para violão, arcadas para violino, uso do pedal para piano e respirações e letra para voz (fig. 5).
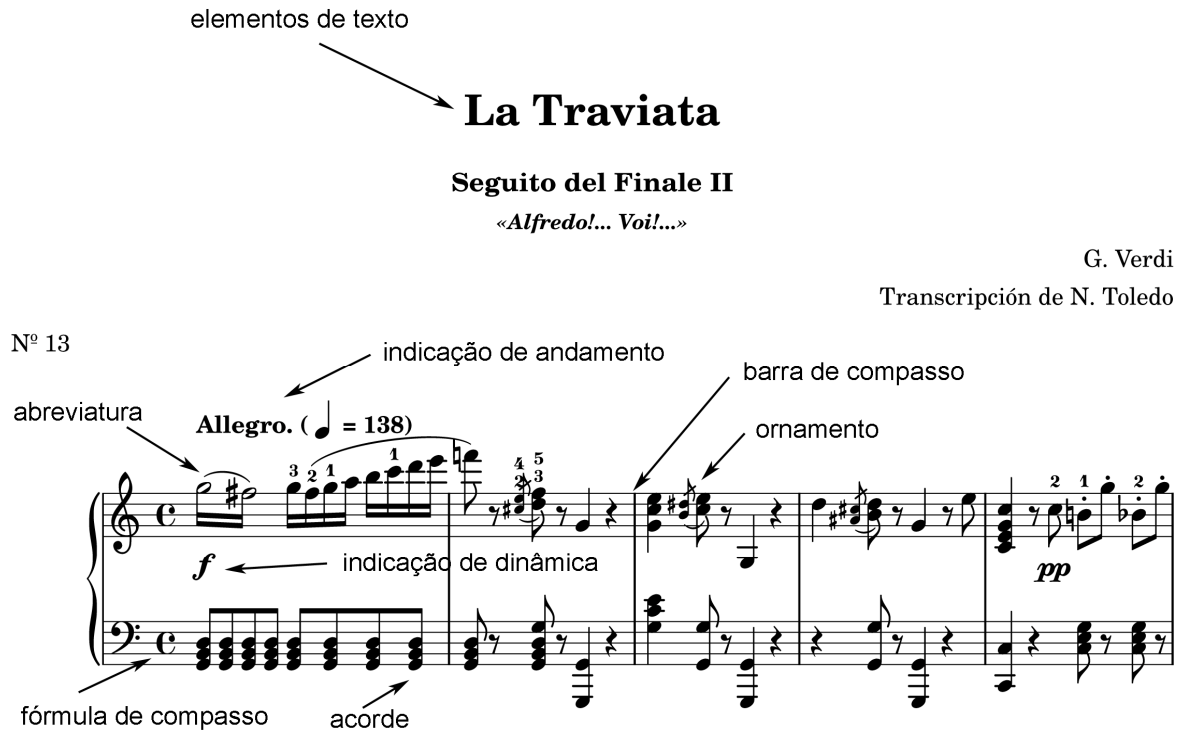
Figura 5 – Trecho de partitura.
Fonte: (Verdi, Mutopia Project)

I.2 - USABILIDADE

A usabilidade de interfaces é um tópico que tem recebido muito destaque, sendo alvo de grandes investimentos por parte da indústria, e de muitas pesquisas na área acadêmica. Em geral, a literatura sobre o assunto está voltada para o desenvolvimento de sistemas e trata a avaliação de interfaces como parte de um processo iterativo de engenharia de usabilidade, em um ciclo em que estas avaliações são um subsídio para as etapas seguintes do desenvolvimento. Este processo pode inclusive começar antes mesmo do início da implementação do sistema, usando outros sistemas (de empresas concorrentes), versões anteriores do mesmo sistema, ou até protótipos desenhados em papel como ponto de partida (NIELSEN, 1993). Este processo iterativo também é recomendado para o desenvolvimento de sites para a internet (KRUG, 2006).

Embora o foco destes trabalhos esteja no desenvolvimento, os conceitos, métodos e ferramentas expostos podem ser aplicados em outras atividades onde seja necessária a avaliação da usabilidade de sistemas. Um exemplo seria realizar uma comparação de diversas alternativas de sistemas em um processo de aquisição (NIELSEN, 1993).

I.2.1 - Atributos de Usabilidade

A usabilidade não é uma propriedade unidimensional das interfaces. Segundo Nielsen (1993), seus múltiplos componentes são normalmente associados a cinco atributos: “*aprendabilidade*”, *eficiência*, “*memorabilidade”*, *erros* e *satisfação*, que serão detalhados a seguir.

I.2.1.1 - “Aprendabilidade”

O sistema deve ser de fácil aprendizado, para que o usuário possa começar a usá-lo de modo efetivo rapidamente. A facilidade de aprendizado é considerada por muitos

como o atributo mais importante, uma vez que a primeira impressão que o usuário tem do sistema é quando está aprendendo a usá-lo. É também talvez o atributo mais fácil de ser medido: basta medir o tempo que um grupo de usuários que nunca usou o sistema leva para adquirir certa proficiência no seu uso.

Como mostrado na (fig. 6), diferentes sistemas possuem diferentes curvas de aprendizado. Um sistema focado em usuários novatos pode ter um rápido aprendizado inicial, mas ser menos eficiente, enquanto um sistema voltado para usuários experientes pode ter um aprendizado mais difícil, mas permitir altos níveis de eficiência para estes usuários. Ao avaliar a usabilidade de um sistema, devemos sempre levar em conta o seu grupo de usuários alvo, pois muitas vezes alguns dos atributos de usabilidade podem ser conflitantes, e é necessário buscar um compromisso de forma a atender da melhor maneira as expectativas dos usuários.

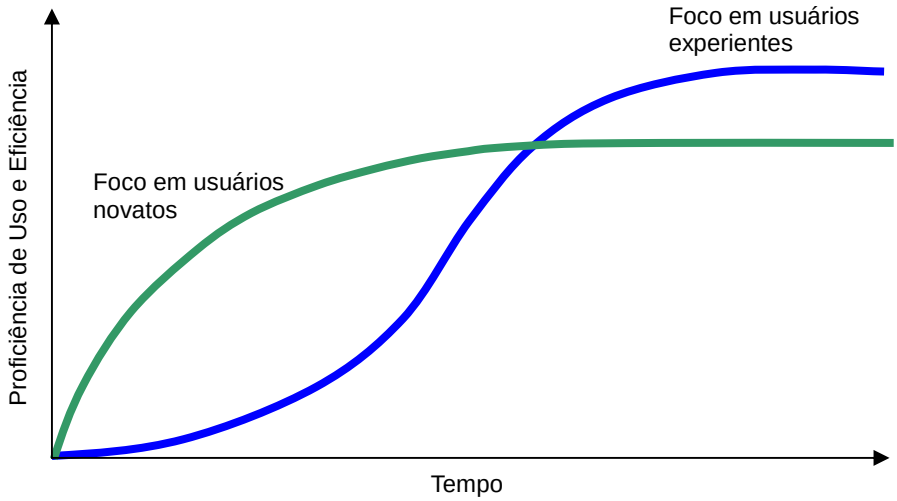

Figura 6 – Curvas de Aprendizado.
Fonte: Nielsen (1993)

I.2.1.2 - Eficiência

O sistema deve proporcionar uma alta produtividade aos usuários experientes -

aqueles que já se encontram na parte plana da curva de aprendizado mostrada na (fig. 6). Para medir este atributo, é necessária a participação de usuários experientes, o que pode ser problemático no caso de sistemas novos (em desenvolvimento). A própria definição de quando alguém é considerado um usuário “experiente” está sujeita a discussões no momento do planejamento da avaliação.

No caso dos softwares de editoração de partituras, podemos considerar que eficiência e produtividade significam a capacidade de produzir em um tempo adequado, partituras completas e corretas (de acordo com as regras da notação e com as práticas de layout mais comuns), esteticamente agradáveis e que permitem ao intérprete uma boa leitura, inclusive para peças mais complexas com maiores dificuldades de notação.

I.2.1.3 - “Memorabilidade”

Além dos usuários novatos e dos experientes, há uma outra grande categoria de usuários, os usuários casuais. Os usuários casuais não usam o sistema com a mesma frequência dos usuários experientes, mas por outro lado não precisam aprender a usar o sistema do início, como os usuários novatos, bastando relembrar como usá-lo baseado em seu aprendizado anterior. Este tipo de usuário usa o sistema não como a sua ferramenta principal de trabalho, mas como uma ferramenta suplementar ou para circunstâncias excepcionais.

Este tipo de usuário se beneficia de uma interface que é fácil de lembrar. Uma interface fácil de aprender normalmente é também fácil de lembrar, mas a rigor são atributos diferentes. A facilidade de lembrar, apesar de importante, é muito difícil de medir e por isso é raramente testada.

Como muitos usuários de programas de editoração de partituras não são profissionais da música, e mesmo entre os que o são, nem todos trabalham com edição de partituras tão frequentemente como copistas, arranjadores ou compositores, os usuários casuais são uma parcela importante do público alvo deste tipo de sistema.

I.2.1.4 - Erros

Em uma boa interface, os usuários cometem poucos erros, e caso cometam erros é fácil recuperar-se deles. Além disto, erros catastróficos (que provocam defeitos no produto final do sistema ou que destroem o trabalho do usuário de maneira difícil ou impossível de se recuperar) não devem ocorrer. Para este atributo, erros são ações do usuário que não resultam no objetivo pretendido, e não erros de lógica ou de codificação do programa (“bugs”).

I.2.1.5 - Satisfação

Os usuários devem gostar do sistema e considerá-lo agradável, e sentirem-se satisfeitos em usá-lo. Este é um atributo subjetivo e costuma ser medido através de questionários.

Numa ferramenta a serviço da arte, a satisfação subjetiva tem uma importância especial, pois pode afetar a própria atividade de criação.

I.2.2 - Categorias de Usuários

Nos itens sobre os atributos de usabilidade, já foram identificadas três categorias de usuários, quanto a sua experiência com o sistema (especificamente) – os novatos, os casuais e os experientes. Podemos ainda classificar os usuários segundo outras perspectivas, como por exemplo, a experiência com o domínio da aplicação, e a experiência com o uso de sistemas de computador de um modo geral. Estas três dimensões definem um espaço (o “cubo de usuários”) onde podemos posicionar um determinado usuário ou grupo de usuários (fig. 7). (NIELSEN, 1993)

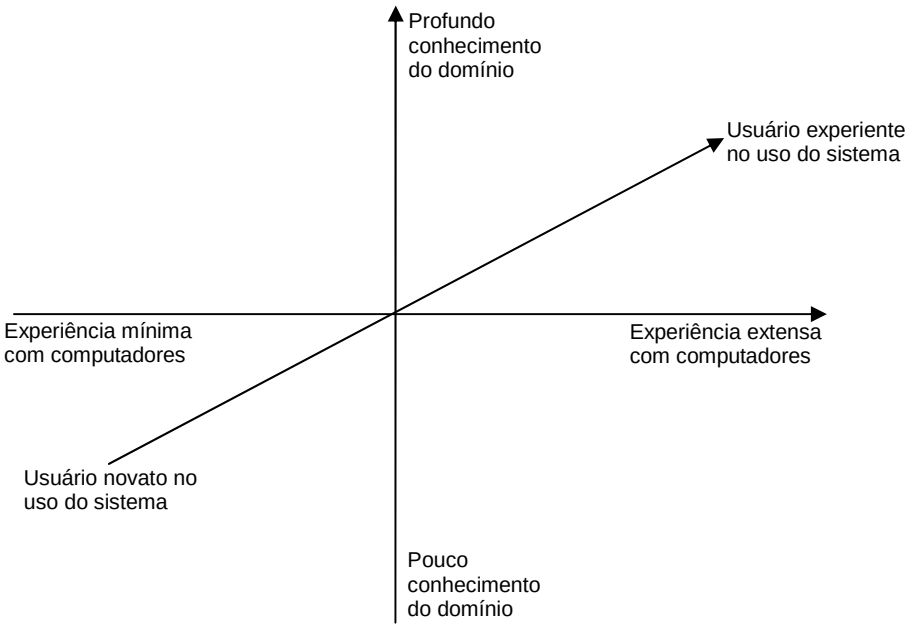

Figura 7 – Cubo de Usuários
Fonte: Nielsen (1993)

O eixo da experiência com o uso do sistema é o eixo que é normalmente considerado quando se fala em “experiência do usuário”. Alguns programas podem ser voltados exclusivamente para usuários novatos, pois normalmente um usuário não os utiliza mais do que poucas vezes (por exemplo, um sistema de balcão de informações em um shopping center). Outros sistemas podem ser voltados unicamente para usuários treinados e experientes. A maioria dos sistemas (como os softwares de editoração de partituras), entretanto, é voltada tanto para novatos como para usuários experientes, e portanto deve ser adequada às necessidades de uso de ambos. Isto requer um esforço no desenvolvimento do sistema de modo a evitar que as funcionalidades avançadas disponibilizadas para os usuários experientes dificultem a compreensão do software por parte dos novatos, e que os recursos de apoio aos novatos sejam transtornos ou empecilhos à produtividade dos usuários experientes. Dentre as soluções frequentemente adotadas estão: disponibilizar aceleradores, como atalhos de teclado, para os usuários experientes, criar grupos de telas distintos (telas resumidas para os novatos e telas extensas para os usuários experientes) e até mesmo a criação de interfaces distintas (por exemplo, com o uso

de “wizards” para os novatos). É importante notar que esta flexibilização da interface deve ser feita com muito cuidado, pois, com a adição de novos elementos, a complexidade da interface está efetivamente aumentando e o resultado pode ser diferente do desejado.

Praticamente todos os sistemas de editoração de partituras permitem a edição utilizando apenas o mouse, menus e manipulação de elementos mostrados na tela como ferramentas ou os próprios objetos que compõem a partitura, o que facilita o uso por novatos. Este modo de uso, entretanto, é pouco eficiente (fazendo uma analogia, seria como tentar redigir um texto em um processador de textos usando apenas o mouse), não sendo muito prático exceto para partituras muito simples (pouca variedade de elementos, muitos elementos repetidos) ou muito pequenas (poucas páginas). Assim, praticamente todos estes sistemas permitem aos usuários mais experientes a edição utilizando o teclado do computador (através de atalhos e teclas de função) ou mesmo através de instrumentos musicais (mais frequentemente teclados eletrônicos) conectados ao computador através de uma interface MIDI ou USB.

Em relação ao conhecimento do domínio da aplicação, uma interface para um usuário especialista pode apresentar uma densidade de informação maior e utilizar uma terminologia específica da área, enquanto uma interface para usuários leigos necessita explicar cada opção, as ações realizadas, os termos utilizados e ser mais restritiva quanto às entradas dos usuários para evitar ações incorretas ou ilegais de acordo com as regras do domínio.

No caso dos softwares de editoração musical, um sistema voltado para usuários com pouco conhecimento de notação musical precisa de uma interface bem diferente de um sistema voltado para usuários fluentes nesta notação e com conhecimentos de teoria musical e das práticas de editoração de partituras. Exceto no caso de sistemas educativos, usados justamente para o ensino da notação musical, não é muito aconselhável o uso de softwares de editoração de partituras por usuários sem nenhum conhecimento deste domínio. Isto implicaria em um esforço de aprendizado dobrado, pois além de aprender uma notação complexa, o usuário precisaria aprender simultaneamente o uso de uma interface em geral extensa e especializada.

Usando novamente uma analogia, seria como tentar alfabetizar um aluno e ensiná-lo a usar um editor de textos ou software de desktop publishing profissionais simultaneamente. Portanto, parece mais adequado aprender ao menos o básico da notação usando lápis e papel, e apenas uma vez “alfabetizado” na escrita musical, tentar editar usando o computador. Na avaliação das interfaces será considerado que o usuário tem ao menos um mínimo de conhecimento do domínio.

O último eixo na (fig. 7) se refere à experiência do usuário com sistemas de computação em geral, e não com o sistema específico sendo analisado. Um usuário com experiência anterior com computadores, pode ter uma facilidade inicial maior, especialmente se a interface segue padrões e estilos de interação comuns em outros aplicativos. Este usuário já tem uma noção de como sistemas de computação se comportam em determinadas situações e dos tipos de recursos normalmente oferecidos por sistemas de computação e como utilizá-los. Além disto, um usuário com um conhecimento extenso de computação pode utilizar recursos avançados de combinação de comandos como a criação de macros e scripts, para um máximo de eficiência no uso do sistema. Um outro fator é que o usuário mais experiente com computadores não tem receio de explorar as opções do sistema, pois tem uma noção do tipo de problemas que pode encontrar e de como resolvê-los, muitas vezes é até um desafio encontrar funções não documentadas ou caminhos alternativos mais eficientes. Já o usuário sem intimidade com o computador muitas vezes fica temeroso de executar ações que não façam parte do roteiro de uso que lhe foi ensinado, e quando algo não previsto acontece reage até mesmo de forma emocional.

Os profissionais de música não necessariamente possuem uma experiência extensa com o uso de computadores, portanto os sistemas avaliados neste trabalho devem levar isto em consideração no projeto de suas interfaces. Por outro lado, com a disseminação do uso da tecnologia na música (que tem até sido incluída como parte do currículo na formação de profissionais de música), e com a própria popularização do uso de computadores entre a população em geral, a média da experiência com o uso de computadores dentre este grupo de usuários tem aumentado cada vez mais. Além disto, entre os músicos amadores, muitos têm uma maior experiência com sistemas de computação, incluindo neste grupo um bom número de profissionais da

área de sistemas e ciências exatas. Deste modo, recursos voltados para usuários com expertise em sistemas de computação podem ser adições bem vindas em softwares de editoração de partituras, desde que não venham em prejuízo dos usuários sem esta experiência.

I.2.3 - Heurísticas de Usabilidade

Nielsen (1993) enumera um conjunto de dez princípios gerais ou heurísticas a serem considerados na avaliação sistemática de interfaces. Segundo ele, estes princípios podem evidenciar uma grande parte dos problemas mais comumente observados em interfaces, podendo ser usados até mesmo por não-especialistas em usabilidade. Outro conjunto de critérios muito utilizado são os oito critérios ergonômicos propostos por Scapin e Bastien (LABIUTIL, 2009).

I.2.3.1- Heurísticas de Nielsen

1. **Diálogo Simples e Natural**: Diálogos não devem conter informação irrelevante ou raramente usada. A interface deve ser o mais simples possível, pois a cada elemento a mais em uma tela corresponde um aumento de complexidade (mais esforço de aprendizado para o usuário, maior dificuldade em achar a informação realmente relevante). A diagramação da tela deve ser usada para enfatizar a organização da informação, de maneira lógica e naturalmente compatível com as tarefas para as quais o usuário utiliza esta informação.

2. **Falar a Linguagem do Usuário**: A interface deve usar termos baseados na linguagem do usuário e na terminologia do domínio da aplicação, e não termos de sistemas. Deve haver um bom mapeamento entre a informação exibida e o modelo conceitual de informação do usuário, por exemplo através do uso de metáforas.

3. **Minimizar a Exigência de Memorização do Usuário**: O usuário não deve precisar se lembrar de informação de uma parte do diálogo com a interface para usar em outra. Os usuários têm mais facilidade em reconhecer do lembrar. Instruções de uso do sistema devem estar visíveis ou facilmente acessíveis.

4. **Consistência**: A mesma ação ou comando deve sempre ter o mesmo efeito, e a mesma informação deve sempre ser apresentada no mesmo local e com a mesma formatação em todas as telas. Isto encoraja o usuário a aprender a usar a interface através da exploração, por se sentir mais confiante e por já deter parte do conhecimento necessário para operar novas partes do sistema. A consistência é um dos princípios de usabilidades mais básicos e mais importantes. Ela não é só relacionada ao desenho de telas, mas também às tarefas e à estrutura das funcionalidades do sistema. Os problemas de inconsistência podem ser entre o sistema e as expectativas do usuário, entre o sistema e outras interfaces às quais o usuário já está acostumado, ou entre partes do próprio sistema. O desenvolvimento apoiado no uso de padrões de interface é uma forma de tentar evitar alguns destes problemas.

5. **Feedback**: O sistema deve manter o usuário continuamente informado sobre o que está fazendo e como está interpretando as entradas do usuário (não apenas no caso de erros, mas também fornecendo feedback “positivo”). O tempo de resposta às entradas do usuário deve ser razoável, e, especialmente no caso de tarefas longas, deve haver uma indicação de progresso ou de tempo restante estimado (além de avisar o usuário sobre o término da tarefa).

6. **Saídas Claramente Marcadas**: O usuário deve se sentir no controle do diálogo com o sistema. Todas as partes do sistema devem fornecer saídas caso sejam invocadas por engano ou caso o usuário mude de idéia. Por exemplo, todos os diálogos devem possuir um botão de “cancela” ou outra facilidade de escape e todo processamento longo deve poder ser interrompido imediatamente. Uma outra forma de saída é a facilidade de “undo” (desfazer) que retorna o sistema a um estado anterior. Com as

facilidades de escape e desfazer, o usuário se sente encorajado a explorar o sistema, pois sabe que pode voltar atrás caso encontre algum problema ao experimentar uma nova funcionalidade.

7. **Atalhos:** mesmo que seja possível operar uma interface apenas com poucos conhecimentos básicos, um usuário experiente deve ter à sua disposição aceleradores que permitam realizar as tarefas mais comuns mais rapidamente ou executar tarefas mais complexas (não acessíveis aos novatos). Alguns exemplos de aceleradores encontrados em interfaces: atalhos de teclado ou teclas de função, gestos de mouse, uso de gabaritos (templates), facilidades de macro e script, auxílio para digitação como auto-correção e auto-completação, reuso do histórico da interação.

8. **Boas Mensagens de Erro**: Mensagens de erro devem ser expressas em linguagem simples (não em códigos), indicar precisamente qual o problema e sugerir soluções de forma construtiva. Elas devem ser polidas e não intimidar ou culpar o usuário. A recuperação de um erro deve ser fácil. O usuário deve ser capaz de desfazer os efeitos do erro e continuar a interação sem precisar repetir o trabalho já feito.

9. **Prevenção de erros**: Melhor ainda do que o princípio anterior, seria ser capaz de evitar qualquer erro em primeiro lugar. Como exemplo, a seleção em uma lista pode evitar erros de digitação. Comandos com consequências graves devem ser precedidos de uma confirmação (“Tem certeza...?”) para evitar que sejam invocados por engano. Deve-se evitar comandos muito parecidos, para que usuário não confunda um com outro. Uma recomendação enfática em (NIELSEN, 1993), é que se evitem *modos*. Um programa que possui modos se comporta de maneira diferente dependendo do modo em que está no momento. Como nem todas as ações estão disponíveis o tempo todo, isto pode ser frustrante para o usuário. O sistema interpreta a mesma ação aparente do usuário de formas diferentes dependendo do modo, o que pode induzir o usuário a erros. Se o uso de modos for inevitável, o sistema ao menos deve explicitar ao usuário de forma bem evidente em qual modo se encontra.

10. **Ajuda e Documentação**: Caso o sistema não seja suficientemente fácil de usar para prescindir de um sistema de ajuda ou documentação, a informação nesse sistema deve ser fácil de localizar, focada na tarefa do usuário, conter passos concretos para resolver o problema do usuário, e não deve ser longa. A qualidade do texto é importante. A informação contida deve ser correta e refletir a versão do sistema efetivamente em uso. Pode ser interessante disponibilizar a documentação em diferentes níveis, como cartões de referência ou gabaritos de teclado (para uso no dia a dia), tutoriais ou introduções (para uso no aprendizado do sistema) e manuais de referência.

I.2.3.2 - Critérios de Scapin/Bastien

Os oito critérios principais são subdivididos em diversos sub-critérios, e para cada sub-critério existe uma lista de recomendações e uma lista de verificação (com um item para cada recomendação) que podem ser usados tanto para avaliação quanto como um guia para o projeto de interfaces (ERGOLIST, 2009). Existe uma grande sobreposição entre estes critérios e as heurísticas de Nielsen.

1. **Condução**: Refere-se às mensagens, alarmes, rótulos, e outros meios disponíveis para aconselhar, orientar, informar, e conduzir o usuário na interação com o computador. É dividido em quatro sub-critérios: Presteza, Agrupamento/distinção entre itens, Feedback imediato e Legibilidade.

2. **Carga de Trabalho**: Relativo a todos os elementos da interface que têm um papel importante na redução da carga cognitiva e perceptiva do usuário, e no aumento da eficiência do diálogo. Está subdividido em: Brevidade (que engloba Concisão e Ações Mínimas) e Densidade Informacional.

3. **Controle Explícito**: Diz respeito ao processamento das ações do usuário pelo sistema e ao controle que os usuários têm sobre o sistema. O critério Controle Explícito se subdivide em: Ações Explícitas do Usuário e Controle do Usuário.

4. **Adaptabilidade**: Capacidade do sistema reagir conforme o contexto, e conforme as necessidades e preferências do usuário. Dividido em: Flexibilidade e Consideração da Experiência do Usuário.

5. **Gestão de Erros**: Mecanismos que permitem evitar ou reduzir a ocorrência de erros, e quando eles ocorrem, que facilitem sua correção. Dividido em: Proteção contra os erros, Qualidade das mensagens de erro e Correção dos erros.

6. **Homogeneidade/Coerência**: O modo como as escolhas na concepção da interface (códigos, denominações, formatos, procedimentos, etc.) são conservadas idênticas em contextos idênticos, e diferentes para contextos diferentes.

7. **Significado dos Códigos e Denominações**: Adequação entre o objeto ou a informação apresentada ou pedida, e sua referência.

8. **Compatibilidade**: Refere-se à adequação entre as características do usuário e das tarefas, por um lado, e a organização das saídas, das entradas e do diálogo do sistema, por outro. Relativa também ao grau de similaridade entre diferentes ambientes e aplicações.

I.2.4 - Métodos de Avaliação de Usabilidade

Dentre os objetivos da avaliação de usabilidade de interfaces estão: constatar, observar e registrar problemas de usabilidade; calcular métricas de eficácia, eficiência e satisfação do usuário; diagnosticar características do sistema que não estejam em conformidade com padrões de usabilidade; prever dificuldades de aprendizado na operação do sistema; prever os tempos de execução das tarefas; conhecer a opinião do usuário sobre o sistema e, sugerir ações de re-projeto (caso a avaliação esteja incluída no ciclo de desenvolvimento do sistema). Com base nestes objetivos, podemos dividir as técnicas de avaliação de interfaces em três grandes

grupos: as técnicas prospectivas, as técnicas preditivas e as técnicas objetivas.

As técnicas prospectivas, como a aplicação de questionários e entrevistas com os usuários, buscam a opinião destes sobre o sistema.

As técnicas preditivas ou diagnósticas buscam prever os problemas do sistema sem a participação do usuário. Estão divididas em técnicas baseadas em modelos formais, como as avaliações analíticas, e baseadas no conhecimento, como as avaliações heurísticas (baseadas no conhecimento e na experiência dos avaliadores) e as inspeções por listas de verificação (que embutem o conhecimento na ferramenta de inspeção).

As técnicas objetivas ou empíricas buscam constatar os problemas a partir da observação do usuário interagindo com o sistema. Estão divididas em ensaios de interação, como os testes de usabilidade, e sistemas de monitoramento como, por exemplo, análise de registros (logs) do sistema em operação.

São apresentadas a seguir algumas destas técnicas com mais detalhes.

I.2.4.1 - Questionários e Entrevistas

Consistem em obter informações sobre a usabilidade do sistema através de perguntas ao usuário. Como não há a manipulação direta da interface, são consideradas técnicas indiretas. Através destas técnicas, pode-se avaliar aspectos difíceis de medir objetivamente, como a satisfação do usuário e suas ansiedades.

Questionários são impressos em papel ou apresentados interativamente em um computador, e podem ser administrados sem a necessidade da presença de um avaliador. Entrevistas, por outro lado, necessitam da presença de um entrevistador que formule as perguntas e registre as respostas do usuário. Os questionários portanto necessitam de menos mão de obra de avaliadores de usabilidade, mas as entrevistas podem ser mais flexíveis, sendo mais apropriadas para estudos exploratórios (onde não se tem certeza sobre que informação é buscada, e o

entrevistador precisa ir se adaptando à situação).

Questionários são interfaces com o usuário e estão sujeitos aos mesmos princípios aplicáveis a outros sistemas. Devem ser fáceis de usar, não devem ser muito longos nem incomodar ou irritar os usuários. Em geral as perguntas são bastante fechadas, mas pode haver espaço para perguntas abertas aonde o usuário redige sua própria resposta em linguagem natural. Alguns dos tipos de solicitação de informações mais usados são: informar um fato ou valor (por exemplo,quantas horas por semana usa o sistema), a escolha em uma lista (que pode conter espaço para alternativas escritas, mas deve ser o mais completa possível) e informar uma opinião em uma escala graduada.

I.2.4.2 - Avaliação Heurística

A avaliação heurística é uma forma de “inspeção de usabilidade”, em que um especialista explora a interface sistematicamente avaliando sua usabilidade. Em contraste com os testes de usabilidade, as inspeções de usabilidade não têm a participação de usuários, apenas de alguns poucos avaliadores que examinam a interface e julgam-na baseados em princípios e critérios reconhecidos (as heurísticas).

Embora uma avaliação heurística possa ser conduzida por apenas um avaliador, o mais aconselhável é que as avaliações de diversos especialistas sejam agregadas, uma vez que normalmente um único avaliador não consegue identificar todos os problemas de usabilidade em uma interface, e cada avaliador detecta um conjunto diferente de problemas. Neste caso, cada especialista deve realizar sua avaliação individualmente, e só depois se reunirem para agregar os resultados.

O resultado de uma avaliação heurística é uma lista de problemas de usabilidade encontrados, associados ao princípio de usabilidade que foi violado em cada caso, na opinião do especialista. Esses problemas são chamados de problemas potenciais, uma vez que alguns deles podem não ser problemas reais, na medida em que eventualmente usuários representativos usando efetivamente o sistema não

sejam afetados por eles (e portanto o esforço de melhoria da interface pode ser melhor aproveitado atacando-se outros problemas).

Segundo Nielsen (1993), o número de problemas encontrados em uma avaliação heurística depende fortemente da experiência dos avaliadores. Ele classifica os avaliadores em “novatos”, com experiência em computadores mas sem nenhum conhecimento especial sobre usabilidade, especialistas “simples”, com experiência em usabilidade mas sem experiência no domínio da aplicação, e especialistas “duplos”, com experiência tanto em usabilidade quanto no domínio da aplicação. O desempenho dos novatos é considerado fraco, dos especialistas simples bem melhor e dos especialistas duplos muito melhor. Assim, mesmo sendo possível realizar avaliações heurísticas com novatos (não havendo disponibilidade de especialistas), o mais recomendado é o uso de especialistas, preferencialmente “duplos”. Não foi considerado por Nielsen o caso particular de um avaliador iniciante em usabilidade, mas com bom conhecimento no domínio da aplicação, que é a situação das avaliações realizadas neste trabalho.

I.2.4.3 - Testes de Usabilidade

Testes com o usuário são a técnica mais fundamental em avaliação de usabilidade, uma vez que a informação obtida diretamente da interação do usuário com o próprio sistema e dos problemas concretos que acontecem é valiosíssima e em grande parte insubstituível. Entretanto, as outras técnicas citadas podem ser usadas como complemento aos testes com o usuário, para obter informações adicionais, e mesmo como formas de avaliação alternativas quando os recursos disponíveis são restritos.

Questões importantes em relação a testes de usabilidade são a confiança ou repetibilidade dos resultados, e a validade das conclusões, o que pode ser estimado através de critérios estatísticos.

Os testes devem ser cuidadosamente planejados, conduzidos e analisados. É necessária uma infra-estrutura para a realização dos testes com instalações adequadas, dispositivos para registro das interações (como câmeras de vídeo),

recrutamento de usuários e tempo disponível para a realização dos testes e análise dos registros. Um teste normalmente consiste em designar uma ou mais tarefas para o usuário e observar e registrar a interação do usuário com o sistema para efetuá-las. Pode ser pedido ao usuário que “pense em voz alta” explicando os seus passos. É aconselhável filmar não só a tela do computador, como também o uso do teclado e do mouse e as reações do próprio usuário.

Pode haver uma outra sala, à prova de som, separada por um espelho falso da sala onde os testes são realizados, para que outras pessoas além do avaliador responsável por aplicar os testes podem acompanhar a sua realização. Geralmente esta sala é reservada aos membros da equipe de desenvolvimento do software, para que estes possam verificar como é a experiência real de uso do seu sistema por usuários representativos. Segundo Spolsky (2001), este é o maior benefício que um teste de usabilidade pode proporcionar, muito mais do que qualquer relatório ou parecer de especialistas, e sem isto o teste seria praticamente inútil. Os desenvolvedores se situam em uma região do “cubo de usuários” normalmente muito diversa da dos usuários, e têm muito conhecimento sobre o funcionamento interno do sistema. Eles costumam ficar surpresos com as dificuldades e reações dos usuários, ao ponto da recomendação sobre a sala de observação ser à prova de som ser fundamental para evitar constrangimentos aos participantes do teste. Após presenciar uma sessão de testes, os desenvolvedores têm a oportunidade de retomar o desenvolvimento com outra perspectiva.

Apesar de ser uma das principais técnicas de avaliação de usabilidade, foi considerada inviável sua aplicação neste trabalho.

I.3 - DETALHAMENTO DO ESCOPO DAS AVALIAÇÔES

Para este trabalho, foram utilizadas duas técnicas de avaliação: um questionário on-line e uma avaliação heurística simplificada dos softwares em estudo. Foram selecionados os softwares Encore, Finale e Sibelius. Softwares sem interfaces gráficas não foram avaliados, embora a opinião sobre eles tenha sido sondada através do questionário.

O questionário foi elaborado usando a ferramenta “Google Forms”, e disponibilizado na internet. Uma cópia do questionário encontra-se no anexo I. Os usuários convidados a responder o questionário foram os membros do fórum de discussões Violão.org, onde uma excelente discussão sobre o tópico de softwares de edição de partituras já estava em andamento (VIOLÃO.ORG, 2009), e alguns músicos, amadores e profissionais, convidados por e-mail.

Para as avaliações heurísticas, a última versão de demonstração dos softwares disponível em suas home pages foi instalada em um computador rodando Windows XP Pro. A simplificação na técnica foi o uso de apenas um avaliador, o autor do trabalho. O conjunto de problemas potenciais de usabilidade encontrados, portanto, é apenas um subconjunto dos problemas que seriam encontrados por uma equipe composta por vários avaliadores experientes, e portanto não é possível concluir qual o software mais problemático, ou que um software é melhor ou pior do que o outro. Entretanto, as avaliações realizadas são válidas como uma ilustração da aplicação da técnica e dos diversos tipos de problemas que ela pode apontar. Além disto, provavelmente os problemas potenciais encontrados estão entre os mais evidentes.

Na avaliação heurística, o cenário considerado foi o uso destes sistemas por um usuário casual, com um bom domínio da notação musical e teoria musical, que edita partituras para uso próprio ou para compartilhar com grupos musicais dos quais participa, e que está avaliando os diferentes softwares com o objetivo de escolher um deles para uso próprio. Não foi avaliado o uso visando à produção de partituras profissionalmente diagramadas com o objetivo de publicação comercial, uma vez que seria necessário o auxílio de um especialista no domínio da aplicação.

CAPÍTULO II – RESULTADOS

II.1 - AVALIAÇÃO HEURÍSTICA

II.1.1 - Encore

A (fig. 8) mostra a tela principal da versão de demonstração do software Encore (versão 5.0.1). Os principais elementos são: uma barra de menus, uma barra de ferramentas logo abaixo da barra de menus, uma janela flutuante onde são exibidas diversas paletas e a tela onde é editada e visualizada a partitura. O uso mais básico é muito intuitivo. Primeiro ativa-se uma das ferramentas na barra de ferramentas, que usa como metáforas símbolos de fácil entendimento (um cursor para “seleção”, uma borracha para “apagar” e um lápis para “escrever”). Caso a ferramenta ativada seja “selecionar”, ao clicar-se em um elemento na partitura, este é selecionado, sendo possível também realizar seleções múltiplas com arraste de mouse e uso de teclas ctrl e shift como em outras aplicações baseadas em janelas gráficas. Caso a ferramenta ativada seja “apagar”, ao clicar-se sobre um elemento na partitura, este é apagado. Caso a ferramenta ativada seja “escrever”, ao clicar-se na partitura, o elemento que estiver selecionado na paleta flutuante é inserido aproximadamente na posição do cursor (é feito um posicionamento automático de acordo com regras de teoria musical e de formatação). Para realizar outras tarefas mais complexas de edição e configuração, usam-se diálogos acionados através dos sub-menus da barra de menus. A distribuição dos sub-menus entre os diversos itens da barra de menus segue uma organização bem intuitiva, sendo fácil descobrir como se realizar uma determinada tarefa. Foram encontrados, entretanto, alguns problemas de usabilidade que serão relatados a seguir.
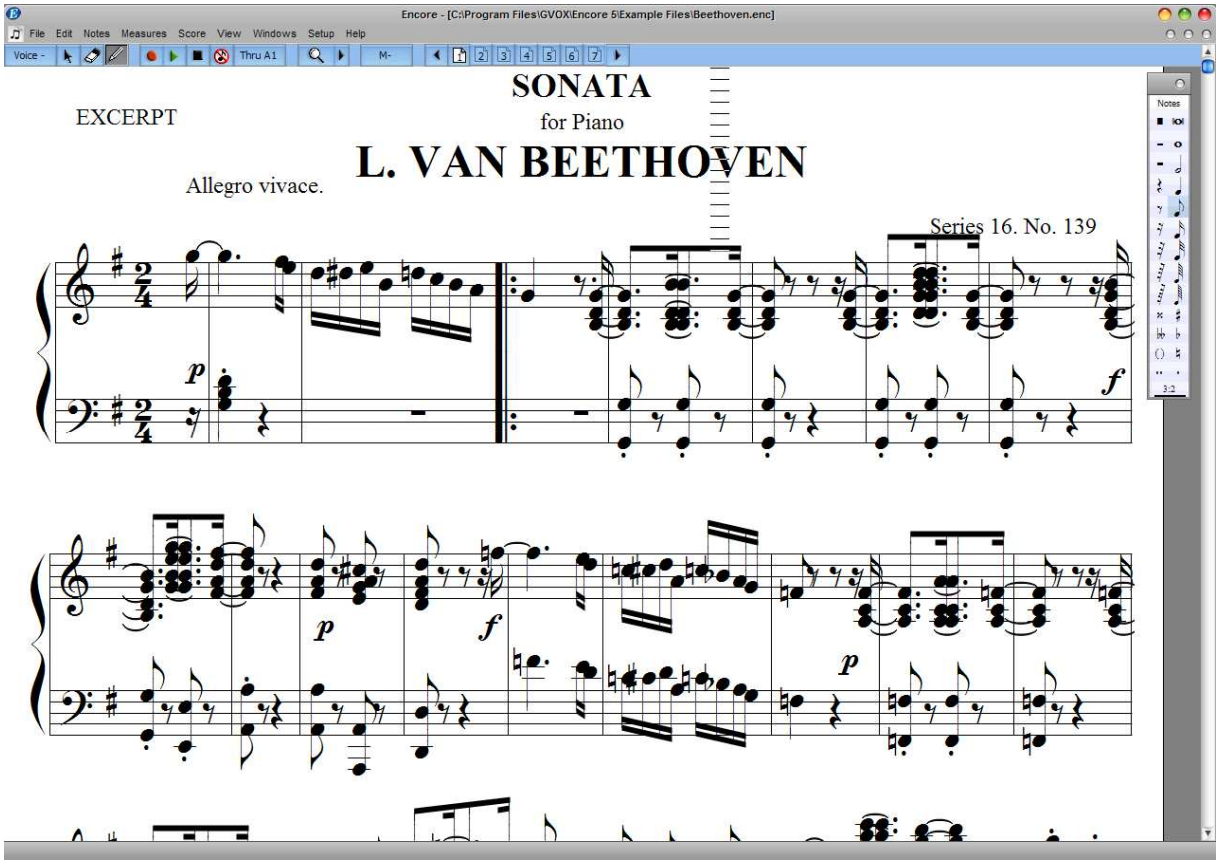
Figura 8 – Tela principal do software Encore.
Fonte: Captura de tela do Software em execução no microcomputador usado para os testes.

A janela flutuante com a paleta é talvez a parte mais importante desta interface. O problema com esta janela flutuante é que não existe nenhuma indicação visual de que ao se clicar no título da paleta, outra paleta é exibida, e assim por diante totalizando as onze paletas mostradas na (fig. 9). Este grande número de paletas permite que partituras bastante elaboradas sejam editadas com facilidade, mas primeiro o usuário precisa “adivinhar” que elas estão acessíveis com um simples clique. Esta é uma violação da heurística “diálogo simples e natural”, que diz que “a diagramação da tela deve ser usada para enfatizar como a informação é organizada, de maneira lógica e naturalmente compatível com as tarefas para as quais o usuário a utiliza”. Pode-se dizer que também há uma violação da heurística “minimizar a exigência de memorização do usuário”, uma vez que não haveria necessidade de memorizar o comportamento desta janela flutuante, se houvesse alguma indicação visual como, por exemplo, setas clicáveis à esquerda e à direita do título da paleta (botões de navegação).

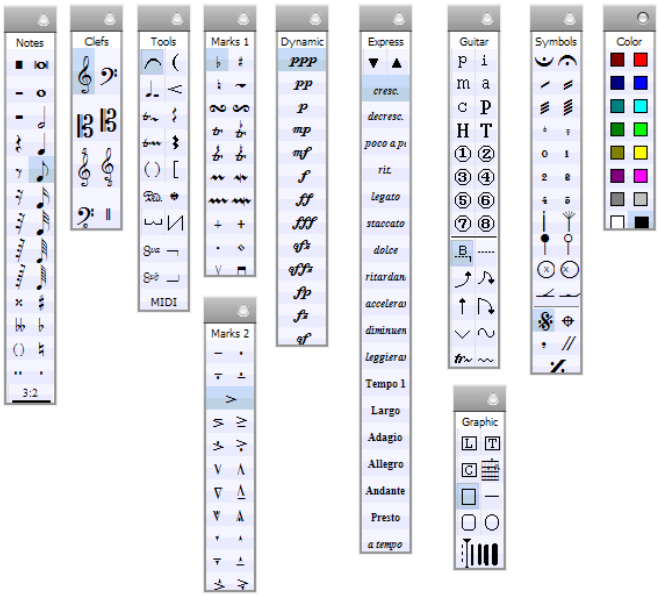

Figura 9 – Paletas de símbolos disponíveis.
Fonte: Montagem elaborada pelo autor a patir de capturas de tela do Software em execução no microcomputador usado para os testes.

Embora a edição usando paletas e o mouse seja simples e intuitiva, não é muito eficiente, a não ser que o mesmo elemento se repita diversas vezes seguidas na partitura. No caso mais geral, é necessário escolher um elemento na paleta, clicar na partitura, voltar à paleta para escolher um novo elemento (talvez necessitando mudar de paleta), clicar na partitura, e assim por diante. Fazendo uma analogia, é como tentar editar um texto usando o mouse e uma paleta com as letras do alfabeto – em geral, usar o teclado é mais rápido. Principalmente para a entrada de notas e pausas, que geralmente são os principais elementos e encontrados em maior quantidade e variedade nas partituras, é desejável a maior eficiência possível. O Encore permite a entrada de elementos na partitura por outros meios além do mouse, que são o uso do teclado do computador e o uso de um instrumento musical conectado ao computador por uma interface MIDI e, portanto, procura seguir a heurística “atalhos”, fornecendo opções mais eficientes para usuários mais avançados. Entretanto, como a implementação da entrada de notas por atalhos de teclado apresenta alguns problemas, podemos dizer que este critério não é plenamente satisfeito. Para inserir notas na partitura utilizando o teclado, é necessário ativar uma janela flutuante com um “piano virtual”, mostrada na (fig. 10),

e então selecionar a caixa “QWERTY Keyboard Note Entry”. Seleciona-se a duração da nota com uma tecla numérica, e a altura da nota com uma tecla da linha “asdfghjkl” do teclado, que representam as teclas brancas de um piano, e algumas teclas da linha “qwertyuiop”, que representam as teclas pretas (fig. 11). Não é possível entrar pausas ou alterações como o sustenido usando apenas o teclado, é necessário retirar a mão do teclado, levar até o mouse, utilizar a paleta, e então voltar para o teclado para continuar a edição. Para mudar a oitava corrente, também é necessário usar o mouse no “piano virtual”.

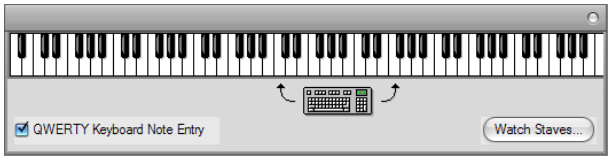

Figura 10 – Piano Virtual
Fonte: Captura de tela do Software em execução no microcomputador usado para os testes.

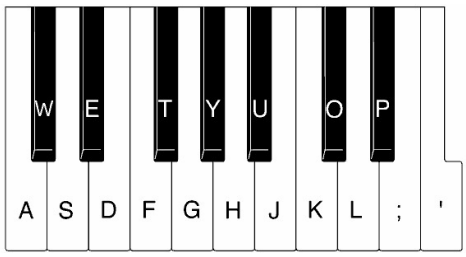

Figura 11 – Layout das notas no teclado QWERTY.
Fonte: Manual do Usuário do Encore (HOTCHKISS, 2009).

As restrições à edição usando o teclado comentadas acima são devidas à violação de uma outra heurística, a “consistência”. Quando a entrada de notas através do teclado QWERTY não está ativa, a tecla “S” é um atalho para inserir um sustenido, e a tecla “R” alterna entre entrada de notas e entrada de pausas. Quando a entrada de notas através do teclado QWERTY está ativa, a tecla “S” insere a nota dó e a tecla “R” fica sem função. Este problema acontece com outros atalhos também; a mesma

tecla tem funções diferentes em situações diferentes.

Não foi avaliada a funcionalidade de entrada de notas usando um instrumento musical conectado ao computador por interface MIDI. Esta funcionalidade exige hardwares específicos (instrumento com interface e computador com uma placa MIDI ou adaptador USB/MIDI). Os instrumentos musicais mais usados para esta função são os teclados eletrônicos (outros tipos instrumentos, como instrumentos de cordas ou sopros, costumam ser mais caros, difíceis de encontrar, ter problemas de performance e exigir uma grande modificação na técnica instrumental do músico), o que exclui esta opção para usuários que não tem fluência em instrumentos de tecla. Fica apenas o comentário de que segundo a documentação, a entrada de notas por MIDI do Encore é a mais básica dos três softwares avaliados, o único recurso para facilitar este modo de entrada é a quantização do ritmo.

Em relação à heurística “atalhos”, outro recurso disponível para agilizar a criação de documentos é o uso de templates (gabaritos). Também existe um wizard para auxiliar a criação de um novo documento, passo a passo. É possível visualizar a partitura de forma “linear”, o que pode ser mais cômodo para a entrada de notas do que visualizar toda a paginação.

Quando é feita a seleção de um elemento usando a ferramenta selecionar e clicando-se sobre este elemento, não há indicação visual alguma de que o objeto foi selecionado. Com a seleção usando o arraste de mouse, uma área fica indicada em vídeo reverso, mas pode não ficar claro se elementos que ficaram com partes de fora desta área foram selecionados ou não. Isto é uma violação da heurística “feedback”.

Existem vários problemas com a ajuda e a documentação. O arquivo de entrada que deveria ser usado no tutorial, por exemplo, não foi instalado, o que prejudica muito a sua função didática. Em vários lugares do manual e da ajuda, as ilustrações e os textos não correspondem às janelas efetivamente exibidas pelo software (fig. 12). A organização e estrutura de navegação do help são confusas, dificultando a busca de informações.

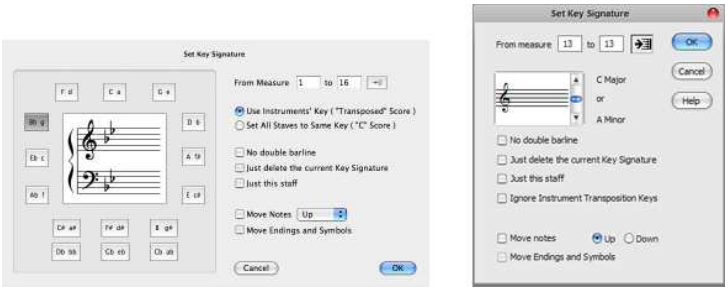

Figura 12 – Diálogo de armadura de tom segundo o manual e efetivamente exibido. O texto do manual também se refere a botões dispostos segundo o círculo das quintas, ausentes no diálogo real. Fonte: Esquerda: Manual do Usuário do Encore (HOTCHKISS, 2009). Direita: Captura de tela do Software em execução no microcomputador usado para os testes.

O Encore permite importar e exportar MusicXML, que é um padrão aberto, baseado em XML, para representação de partituras, e que pode ser usado para intercâmbio de partituras entre usuários de softwares diferentes, ou no caso de migração para outro software.

Um problema com a heurística “falar a linguagem do usuário”, é que não há informações no site sobre versões em outra língua que não o inglês. Como o vocabulário referente à notação musical e à teoria musical é muito específico, mesmo usuários com um bom domínio de inglês podem ter dificuldades com o software. Existe um manual traduzido para português de uma versão antiga, mas precisa ser adquirido separadamente. Alguns sites na internet oferecem “patches” que realizam a tradução de algumas versões mais antigas do software, mas esta solução não é oficial e pode inclusive representar um problema de segurança.

Outro problema que pode ser notado observando a (fig. 8) é que, com partituras mais complexas, o resultado exibido na tela mostra inúmeros defeitos quanto à diagramação e disposição dos diversos elementos na partitura, com diversos problemas tanto estéticos quanto funcionais, como a colisão de elementos. Embora isto seja ainda mais grave do que apenas um problema de usabilidade, podendo ser considerado como um defeito do software (bug), a usabilidade também é afetada, uma vez que o usuário não pode utilizar diretamente os resultados produzidos pelo algoritmo de diagramação do programa e precisaria manipular manualmente praticamente cada elemento para conseguir um resultado apresentável, o que

demandaria muito tempo e esforço. Isto ainda é mais agravado ao se levar em conta que o resultado impresso não é perfeitamente idêntico ao mostrado na tela, e pode tanto ser necessário imprimir o documento diversas vezes até conseguir o resultado pretendido, quanto gastar tempo e esforço corrigindo problemas que eventualmente não afetariam a impressão. É importante notar que a partitura da (fig. 8) é um dos exemplos fornecidos com o software, que costumam ser preparados de forma a ressaltar todas as virtudes do software, de modo a causar uma primeira boa impressão.

Além disto, nos testes, o software se mostrou instável, apresentando diversos travamentos. Muitas vezes a execução era simplesmente terminada, e o software não possui mecanismos de recuperação do trabalho que estava sendo realizado quando é reiniciado (o que, apesar de não estar relacionado a erros do usuário, afeta a usabilidade, podendo ser entendido como uma violação das heurísticas relativas a erros).

II.1.2 - Finale

A (fig. 13) mostra a tela principal da versão de demonstração do software Finale 2010 (versão 2010.r6). Os principais elementos são uma barra de menus, a tela onde é editada e visualizada a partitura, e várias paletas de ferramentas, usadas para organizar as diversas ferramentas disponíveis em grupos. As paletas que são exibidas podem ser configuradas através dos menus da barra de menus. Qualquer destas paletas pode ficar flutuante, ou ser fixada abaixo da barra de menus, do lado esquerdo ou do lado direito da janela principal.

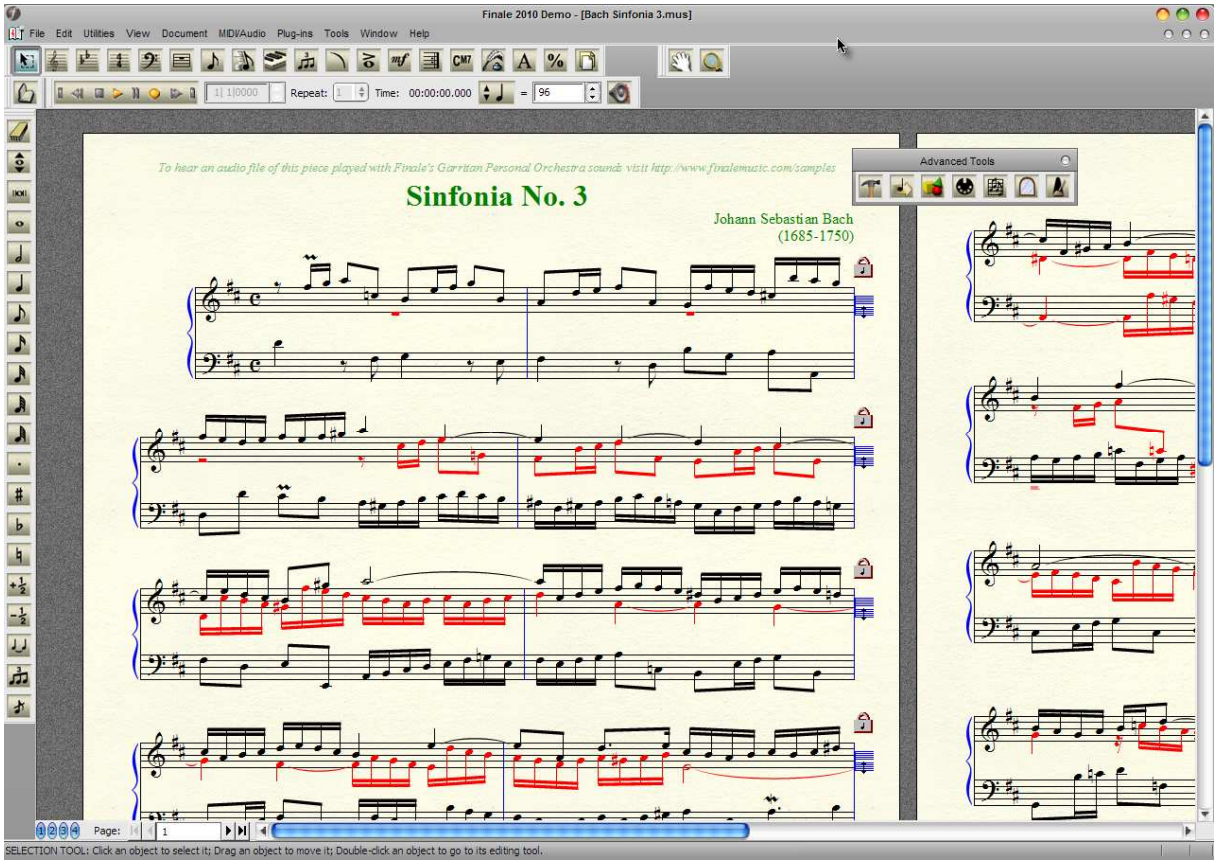
Finale 2010 Demo - [Bach Sinfonia 3.mus]
File Edit Utilities View Document MIDI/Audio Plug-ins Tools Window Help
Repeat: Time: 00:00:00.000
Advanced Tools
To hear an audio file of this piece played with Finale's Garritan Personal Orchestra sounds visit http://www.finalemusic.com/samples
Sinfonia No. 3
Johann Sebastian Bach
(1685-1750)
Page:
SELECTION TOOL: Click an object to select it; Drag an object to move it; Double-click an object to go to its editing tool.

Figura 13 – Finale 2010.
Fonte: Captura de tela do Software em execução no microcomputador usado para os testes.

Esta estrutura do Finale se opõe radicalmente à recomendação de se evitar modos (segundo a heurística “prevenção de erros”). Cada ferramenta que compõe cada uma das paletas coloca o programa em um modo diferente. Existem paletas que apresentam até mais de um modo para a realização da mesma tarefa, e nestes diferentes modos, o mesmo atalho pode funcionar de forma diferente (violação da “consistência”). Como o funcionamento do programa é totalmente baseado em modos, supõe-se que os seus desenvolvedores não considerem isto um problema de usabilidade, ou esperem que a flexibilidade obtida com seu uso compense os aspectos negativos. Este é um problema em potencial de difícil solução, pois seria necessário reescrever praticamente toda a interface, e além do esforço de desenvolvimento envolvido, existe uma grande base de usuários já habituados a esta interface que seria penalizada com uma mudança deste porte.

Alguns dos menus da barra de menus estão sempre disponíveis. Outros são

exibidos apenas em alguns modos, de acordo com a ferramenta selecionada. Isto permite manter a barra de menus com um número pequeno de itens, mas torna o software mais difícil de aprender e de lembrar e mais propenso a erros.

Para a entrada de notas e pausas, existem três modos: o “Simple Entry”, o “Speedy Entry” e o “Hyper Scribe”.

O Simple Entry é o modo de entrada mais voltado para usuários menos experientes. Ele permite a entrada de notas através do teclado ou da escolha de elementos com o mouse em uma paleta de notas ou em uma paleta de pausas, separada. A paleta de pausas não é exibida por default, quando o programa é instalado, devendo ser ativada através de um menu. Os menus mais intuitivos para esta tarefa seriam “View”, ou ainda o menu “Simple” (que só é exibido na barra de menus quando o software está no modo “Simple Entry”), mas a opção se encontra no menu “Window” (violação da heurística “diálogo simples e natural”). A entrada de notas no modo Simple Entry usando o teclado baseia-se no sistema de cifras para associar uma letra de A a G às notas de lá até sol. Isto pode ser um problema para usuários de países onde o uso deste sistema não é comum, ou onde o sistema usado é um pouco diferente (violação da heurística “falar a linguagem do usuário”).

O modo Speedy Entry é voltado para usuários mais experientes. Ele permite a entrada de notas e pausas através do teclado ou de um instrumento musical com interface MIDI. O nome Speedy Entry não é muito justificado, pois as sequências de teclas associadas a cada tarefa têm o mesmo comprimento que as sequências equivalentes no Simple Entry. Neste modo, o teclado QWERTY é mapeado como um teclado de piano, e é possível inserir pausas, modificar oitavas, e realizar qualquer outra tarefa de edição utilizando apenas o teclado, efetivamente buscando atender a heurística “atalhos”. Entretanto, como cada tecla ou combinação de teclas neste modo equivale a notas ou aciona comandos diferentes do modo Simple Entry, existe uma violação da heurística “consistência”.

O modo Hyper Scribe é mais voltado para a entrada utilizando instrumentos musicais com interface MIDI, embora também suporte o uso do teclado do computador. Por este motivo este modo não foi avaliado, entretanto vale a pena ressaltar um recurso

interessante descrito na documentação. É possível usar um sinal (que pode ser, por exemplo, um pedal ligado ao instrumento, ou uma nota específica pré-configurada) para marcar os tempos de um compasso ao mesmo tempo em que se executa o trecho musical que está sendo registrado. Este recurso facilita o uso deste meio de entrada mesmo por usuários que não sejam proficientes no instrumento musical utilizado, compensando uma provável irregularidade rítmica.

Para inserir um compasso através do menu não é usada a ferramenta “measure”, e nem a opção se encontra no menu “Measure” (que não é um menu “permanente”) – ela está no menu “Edit”. Neste menu, existem duas opções: “add measures”, que acrescenta compassos no final da partitura, e “insert measure stack”, que permite acrescentar compassos em qualquer ponto da partitura. Estas são violações da heurística “diálogo simples e natural”. Existem vários outros casos de violação desta heurística, com a alocação pouco intuitiva (ou mesmo não lógica) de opções nos menus.

O arraste de uma seleção usando o mouse, faz uma cópia ao invés de mover a seleção (inconsistência com o funcionamento padrão de outras aplicações).

O diálogo de fórmulas de compasso possui duas barras deslizantes para configuração da fórmula. Manipulando estas barras deslizantes, é possível chegar à mesma fórmula, mas com um número de tempos por compasso diferente, e consequentemente, um grupamento de notas diferente. Como as regras da teoria musical determinam o número de tempos em cada fórmula de compasso, seria mais adequado se o diálogo permitisse apenas interpretações válidas das fórmulas. Se por algum motivo se desejasse um agrupamento de notas fora do padrão, em um ou mais compassos, isto poderia ser feito posteriormente através de uma ferramenta de grupamento de notas, e não neste diálogo. Assim, este diálogo viola as heurísticas “diálogo simples e natural”, “falar a linguagem do usuário” e “prevenção de erros”.

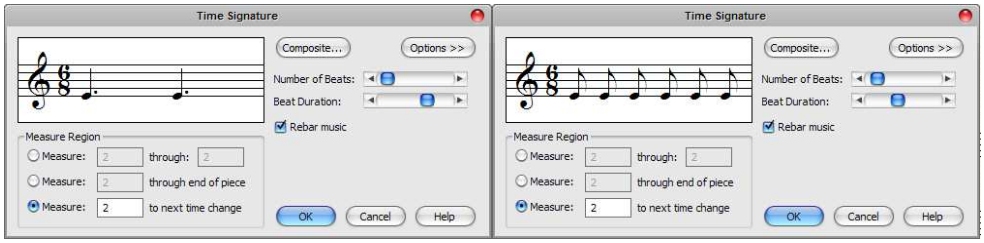

Figura 14 – Diálogo de fórmulas de compasso permite duas configurações diferentes para uma mesma fórmula.
Fonte: Captura de tela do Software em execução no microcomputador usado para os testes.

Não há informações no site sobre versões em outra língua que não o inglês (violação da heurística “falar a linguagem do usuário”). Alguns sites na internet oferecem “patches” que realizam a tradução de algumas versões mais antigas do software, mas esta solução não é oficial e pode inclusive representar um problema de segurança.

A edição de letras, para música cantada, é bem versátil, sendo possível até a importação de um arquivo de texto pré-formatado - de modo a permitir que o software automaticamente associe cada sílaba à(s) nota(s) correta(s).

A edição de cifras de acordes não segue um padrão muito comum, e exige que o usuário memorize as regras de formação das cifras segundo este padrão (não disponibiliza recursos de auxílio para evitar “erros” – cifras fora do padrão). Há uma ferramenta para criação de desenhos de acordes no braço do violão, mas que não funciona com qualquer acorde válido. Se estas ferramentas podem ser personalizadas, as opções de personalização não são facilmente acessíveis (não foram encontradas durante a avaliação). Há, portanto, violação das heurísticas “falar a linguagem do usuário” e “prevenção de erros”.

A reprodução, por default, sempre começa do início da partitura. Para escolher um ponto de início qualquer ou começar de onde parou a última execução, é necessária uma sequência de comandos em diversos menus. Este comportamento é uma violação de “diálogo simples e natural” e “minimizar a exigência de memória do usuário”.

A ferramenta resize permite redimensionar uma nota para tamanhos diferentes do padrão. Isto é útil, por exemplo, em escrita polifônica quando se quer ressaltar qual a voz principal no momento. Heitor Villa-Lobos costumava escrever assim, embora infelizmente nem sempre suas composições tenham sido editadas comercialmente seguindo esta convenção (é um processo de escrita trabalhoso). No Finale, quando uma nota está redimensionada, qualquer ação que dependa do clique do mouse sobre ela, exige que o clique se dê exatamente onde a nota no tamanho original estaria, violando as heurísticas “feedback” e “prevenção de erros”.

É possível extrair partes separadas de uma grade (partitura contendo diversos instrumentos, usada por regentes e arranjadores). As edições feitas na grade se refletem automaticamente nas grades separadas (uma para cada instrumento), assim como as edições em cada parte separada são refletidas automaticamente na grade. Isto facilita a edição, tanto da grade quanto das partes, e evita erros de inconsistência entre a grade e as partes.

O software é extremamente flexível e há vários recursos de personalização, como a criação de macros (chamadas de “Metatools”), templates, estilos de documentos e bibliotecas de elementos personalizados. Há wizards para agilizar a criação de novos documentos. Provavelmente, uma parte dos potenciais problemas de usabilidade encontrados seja devido a um compromisso de projeto em que se priorizou a flexibilidade acima de outros atributos.

O sistema de ajuda e a documentação são bem estruturados, acessíveis e bastante completos. O tutorial é extenso e bem detalhado, cobrindo todas as tarefas mais comuns na criação de uma partitura, de um modo lógico e fácil de acompanhar. O software não é fácil de aprender por exploração, mas um usuário que siga o tutorial estará preparado para utilização eficiente da maioria dos recursos. Entretanto, um usuário casual precisaria recorrer novamente à documentação para se sentir seguro após um período sem utilizar o software. Existem recursos de ajuda como máscaras de teclado que podem ser adquiridos separadamente.

O Finale permite importar e exportar MusicXML.

A qualidade do resultado final é muito boa, embora não seja perfeita – ainda ocorrem problemas, como colisões de elementos, em partituras muito complexas, exigindo alguma intervenção manual na diagramação dos elementos.

II.1.3 - Sibelius

A (fig. 15) mostra a interface do Sibelius. No alto, uma barra de menus. Logo abaixo, uma barra de ferramentas, com os comandos mais utilizados. Ao final desta barra, temos um conjunto de botões “toggle” que controlam a exibição de janelas flutuantes. As janelas flutuantes ativadas por default, mostradas na figura, são uma janela de controle da reprodução da partitura (execução da música escrita usando um sintetizador), uma janela de navegação e uma janela de “teclado numérico” (keypad). Todas as janelas flutuantes são semitransparentes e a partitura é desenhada em uma tela imitando textura de papel sobre um pano de fundo. O resultado visual é muito agradável, porém exige um computador poderoso. Mesmo usando um computador potente, ainda foi possível sentir que o programa é mais pesado do que os outros testados.

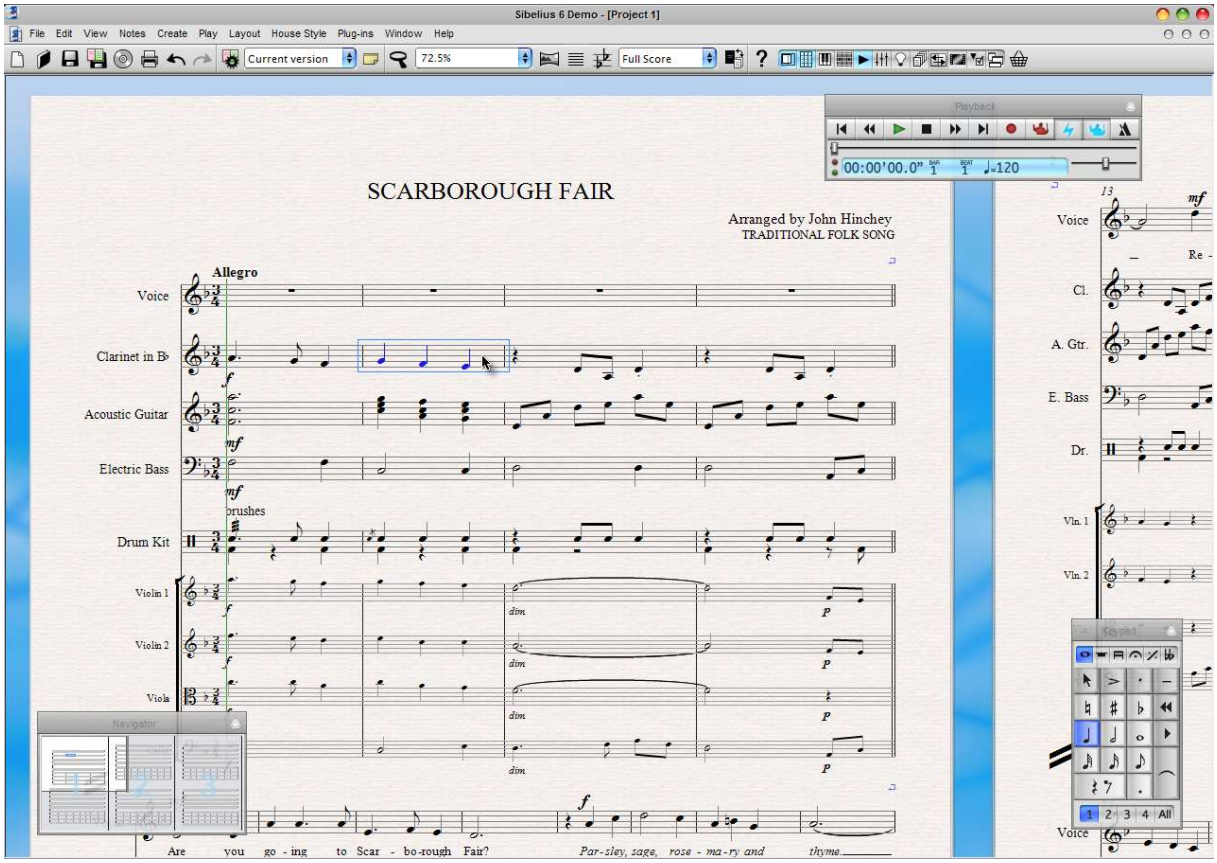

Figura 15 – Janela principal do Sibelius.
Fonte: Captura de tela do Software em execução no microcomputador usado para os testes.

A janela de navegação mostra miniaturas numeradas das páginas da partitura, e utilizando o mouse é possível navegar para qualquer página (paradigma “pan and scan”). Este é um meio bastante intuitivo e prático de navegação. Entretanto, não há nenhuma indicação sobre o número total de páginas, e em uma partitura com muitas páginas, não há como o usuário sabê-lo, e nem se o conjunto de miniaturas exibido no navegador representa a partitura completa ou apenas algumas das páginas. A heurística violada é “feedback”.

A janela “keypad” (fig. 16) é a janela usada para entrada dos principais elementos da partitura (por exemplo, notas e pausas), quando se usa o mouse. Basta selecionar o elemento desejado e clicar no local apropriado na partitura. A fileira de botões de cima seleciona uma dentre seis abas, cada aba contendo um conjunto de elementos diferente. O botão referente à aba em exibição fica realçado. Este esquema permite

disponibilizar um grande número de elementos para uso com o mouse, mas sem exibi-los simultaneamente, economizando espaço em tela. Isto também facilita ao usuário a busca por um determinado elemento, desde que a divisão dos elementos por cada aba seja lógica e intuitiva.

Figura 16 – Várias abas da janela “keypad”.
Fonte: Montagem elaborada pelo autor a patir de capturas de tela do Software em execução no microcomputador usado para os testes.

Este esquema é semelhante às paletas do Encore, porém melhor implementado, uma vez que existe uma indicação visual da presença de outras abas, e uma dica visual do conteúdo de cada aba (o símbolo no botão). Infelizmente, a metáfora da fileira de botões não representa o sistema de abas de maneira tão explícita (outros softwares, por exemplo, simulam abas de fichas de papel, o que é mais intuitivo). Apesar deste detalhe, percebe-se uma preocupação com a heurística “diálogo simples e natural”.

Mesmo para o usuário que prefira fazer a edição usando o teclado ao invés do mouse, a janela “keypad” é muito interessante. O layout desta janela replica o layout do teclado numérico do computador. Assim, a todo o momento, tem-se uma referência visual da função de cada tecla, de acordo com a aba selecionada. Por exemplo, com a primeira aba (“notas”) selecionada, a tecla “4” do teclado numérico seleciona a figura de tempo semínima para a inserção de notas. Pode-se trocar de aba utilizando-se as teclas de função F7 a F12, segundo a mesma ordem da fileira

de botões na janela “keypad”. As teclas e combinações de teclas são semelhantes às usadas no Finale, porém esta indicação visual facilita o uso do teclado para iniciantes, e permite ao usuário casual voltar a utilizar o software sem necessidade de consultar a ajuda e documentação. Na verdade, esta indicação visual já é um tipo de ajuda, e é positiva em relação às heurísticas “minimizar a exigência de memória do usuário”, “atalhos”, “prevenção de erros” e “ajuda e documentação”.

A entrada de notas usando o teclado se baseia no sistema de cifras para associar uma letra de A a G às notas de lá até sol. Isto pode ser um problema para usuários de países onde o uso deste sistema não é comum, ou onde o sistema usado é um pouco diferente. Não há informações no site sobre versões em outra língua que não o inglês. Estas são violações da heurística “falar a linguagem do usuário”.

Existe também um modo de entrada de notas QWERTY (neste caso os atalhos que usam letras do teclado ficam desabilitados).

O Sibelius possui um controle de versões embutido - é possível salvar várias versões de um mesmo documento e voltar a uma das versões anteriores, se for necessário. Este recurso é interessante do ponto de vista de recuperação de erros.

O comportamento das teclas control e shift para fazer e ampliar seleções depende do que já estiver marcado. Não é possível, como em outras aplicações gráficas, fazer uma seleção com arraste de mouse (o arraste faz um “pan and scan” da partitura). Estas são violações da heurística “consistência”.

Assim como nos outros softwares testados, é possível “esconder” um elemento. Este recurso permite inserir um elemento que não será visível na impressão final, mas está presente na partitura. Isto é útil em situações especiais em que não desejamos que a partitura contenha o elemento, mas este é exigido pelo programa para satisfazer alguma regra de teoria musical. Apesar de não ser impresso, um elemento escondido deve ser exibido na tela em uma cor esmaecida, para permitir sua edição. No Sibelius, entretanto, um elemento escondido só é visível enquanto está selecionado, o que fere as heurísticas “feedback”, “prevenção de erros” e “minimizar a exigência de memória do usuário”.

Novamente, as funções de entrada através de interface MIDI não foram avaliadas, sendo apenas comentadas as descrições das funcionalidades constantes no manual. O Sibelius possui uma função “Flexi-Time” que, de modo semelhante ao Hyper Scribe do Finale, é voltada para quem não tem fluência técnica suficiente para garantir uma regularidade rítmica na gravação de cada trecho. O Flexi-Time não necessita que o usuário marque o tempo, como no Hyper Scribe. Ele tenta estimar e compensar automaticamente as variações rítmicas do que está sendo tocado. A funcionalidade de quantização rítmica de um trecho já gravado é acessada através do submenu “Plug-ins | Simplify Notation | Rennotate”, uma localização um tanto escondida para uma função tão comum (violação da heurística “diálogo simples e natural”).

Ao iniciar a reprodução com o botão de play na janela flutuante (ou usando a barra de espaço), a execução começa do início ou de onde parou a última execução. Para escolher um ponto de início qualquer, seleciona-se uma nota e tecla-se “p” (e não a barra de espaço ou botão de play). Este comportamento pode confundir o usuário (violação de “diálogo simples e natural” e “minimizar a exigência de memória do usuário”).

É possível extrair partes separadas de uma grade e as edições feitas na grade se refletem automaticamente nas grades separadas, assim como as edições em cada parte separada são refletidas automaticamente na grade.

Existe uma funcionalidade “Filter”, que, se ativada, permite selecionar apenas objetos de um único tipo. Isto pode ser útil para agilizar a edição e evitar erros.

O acesso à funcionalidade de edição de letras é através de uma seqüência longa de menus ou atalho de teclado – não há uma ferramenta na tela (violação de “diálogo simples e natural”). Se for cometido um erro na divisão de sílabas entre as notas, é preciso apagar do erro em diante e reescrever (violação de “boas mensagens de erro”). É possível importar um arquivo texto, mas não há marcadores – a separação de sílabas é feita automaticamente (violação de “prevenção de erros” e do critério de que o usuário deve estar sempre no controle).

As expressões dinâmicas são consideradas texto, e também não há uma ferramenta na tela, só acesso via menus ou atalho de teclado (violação de “diálogo simples e natural”). Há uma galeria de expressões pré-definidas para escolher (o que ajuda a prevenir erros).

Também as indicações de andamento são consideradas texto, e não possuem uma ferramenta na tela, só acesso via menus ou atalho de teclado (violação de “diálogo simples e natural”). É possível colocar duas indicações simultâneas e conflitantes por engano (violação de “boas mensagens de erros” e “prevenção de erros”). Neste caso, na hora da execução a indicação na janela playback mostra um andamento, mas o andamento real pode ser o outro (violação de “feedback”).

Não existe um diálogo para escolher ou montar cifras de acordes, é necessário digitá-los (violação de “diálogo simples e natural” e “prevenção de erros”).

No Sibelius, é sempre possível selecionar um objeto e realizar operações sobre ele. Não é necessário mudar de modo como no Finale.

O sistema de ajuda não é instalado com a versão de demonstração. Apenas uma parte do tutorial é instalada com a versão de demonstração, e esta primeira parte cobre o software apenas superficialmente. É possível acessar as outras partes do tutorial no site do Software, mas não é possível o download dos arquivos de exemplo associados a elas. O tutorial tem uma estrutura confusa, nem centrada em tarefas, nem em funções. A funcionalidade básica do software é relativamente fácil de aprender por exploração, e a interface oferece segurança a um usuário casual após um período sem utilizar o software. É recomendado o estudo dos manuais para aproveitar os recursos mais avançados. Existem recursos de ajuda como máscaras de teclado que podem ser adquiridos separadamente.

II.2 - QUESTIONÁRIO

Foi publicado um questionário utilizando a ferramenta Google Forms (anexo I). Entre 30 de outubro e 4 de dezembro de 2009, treze pessoas responderam a este questionário, a maioria usuários do fórum de discussão na internet “Violão.org”. Este número é pequeno para obter alguma tendência ou resultado estatístico significante, mas é possível extrair algumas conclusões interessantes sobre a usabilidade dos softwares pesquisados e sobre a percepção que este grupo de usuários tem de cada um. As respostas às perguntas abertas, especialmente, apontaram alguns aspectos importantes, que não foram detectados na análise heurística (realçando o valor da experiência de usuários reais, com tarefas concretas, para a avaliação da usabilidade). No restante desta seção, serão apresentados os resumos das respostas para cada item do formulário, acompanhados de comentários e análises.

As perguntas de 1 a 10 procuram levantar um perfil dos usuários de cada software. As perguntas de 11 a 18 investigam a opinião destes usuários em relação à usabilidade de cada software. As perguntas 19 e 20 são perguntas abertas com liberdade para os usuários se expressarem a respeito dos softwares e do próprio questionário.

As observações sobre o Lilypond e cada software citado em “outros”, são baseadas na opinião de praticamente um único usuário, na maioria das perguntas, e, portanto a ressalva quanto à significância destes dados é ainda mais pertinente.

O anexo II apresenta os dados tabulados, na forma de gráficos de barra e de pizza.

Tabela 1 – Respostas para a pergunta 1.

1. Você usa ou já usou algum software de edição de partituras?	número de respostas	procentagem
Não	0	0,00%
Encore	10	76,92%
Finale	9	69,23%
Lilypond	3	23,08%
Sibelius	6	46,15%
Outros	7	53,85%
Total	13	

Pergunta 1 (tab. 1).

- Opções disponíveis: Não, só lápis e papel; Encore; Finale; Lilypond; Sibelius; Outros. Foi permitido marcar mais de uma opção, e havia espaço para especificar softwares não listados em “outros”.
- O Lilypond, apesar de não considerado na avaliação heurística, foi incluído no questionário, como forma de investigar a opinião de usuários sobre softwares sem uma interface interativa.
- Softwares citados em “outros”: abcm2ps (2); Musescore (2); NoteEdit(1); GNU Denemo (1); Rosegarden (1); TuxGuitar (1). Todos os softwares citados são softwares livres. O abcm2ps utiliza um arquivo de texto com comandos (como o Lilypond). O Musescore e o NoteEdit possuem interfaces gráficas WYSIWYG. O Denemo é uma interface gráfica para o motor do Lilypond. O Rosegarden é um software DAW (Digital Audio Workstation) que possui uma funcionalidade de edição de partituras e o TuxGuitar é um editor específico, voltado para guitarra e instrumentos de corda dedilhada, com uma grande ênfase na edição de tablaturas – portanto, ambos estão fora do escopo deste trabalho.
- Dos treze usuários, dez já utilizaram ao menos dois softwares diferentes. Cinco já utilizaram os três softwares analisados (Encore, Finale e Sibelius). Dos três que só utilizaram um software, dois são usuários do Encore e um do Finale.

Tabela 2 – Respostas para a pergunta 2.

2. Dentre os softwares que você marcou qual é o seu preferido?	número de respostas	porcentagem
Encore	5	38,46%
Finale	2	15,38%
Lilypond	1	7,69%
Sibelius	1	7,69%
Lápis e papel	1	7,69%
Nenhum	1	7,69%
Outros	2	15,38%
Total	13	

Pergunta 2 (tab. 2).

- Um usuário disse preferir lápis e papel
- Um usuário disse não ter preferência.
- Um usuário prefere o Lilypond, e dos “outros”, um prefere o abcm2ps e um prefere o Musescore e o TuxGuitar, totalizando três usuários que preferem softwares livres e dois que preferem softwares com interfaces não interativas.
- Cinco usuários preferem o Encore, dois preferem o Finale e um prefere o Sibelius.

Tabela 3 – Respostas para a pergunta 3.

3. E qual o que você usa mais?	número de respostas	porcentagem
Encore	5	38,46%
Finale	3,5	26,92%
Lilypond	1	7,69%
Sibelius	0,5	3,85%
lápis	1	7,69%
outros	2	15,38%
total	13	

Pergunta 3 (tab. 3).

- Em geral, os usuários marcaram a mesma opção da pergunta 2.
- O usuário que disse não ter preferência diz usar mais o Finale.
- O usuário que diz preferir o Sibelius disse usar mais “Sibelius/Finale”. Dividi o voto dele pelos dois softwares na tabela acima para manter o total em 100%.

Tabela 4 – Respostas para a pergunta 4.

4. Quais os principais motivos pelos quais você usa mais este software?	número de respostas	porcentagem
A qualidade do resultado final.	4	30,77%
É mais rápido.	3	23,08%
É o que eu sei usar melhor.	7	53,85%
Meu trabalho exige a partitura neste formato.	0	0,00%
É o único ao que eu tenho acesso.	4	30,77%
Porque é mais fácil conseguir ajuda de amigos quando tenho alguma dúvida.	1	7,69%
Foi o primeiro que eu aprendi a usar e não quero experimentar outro para não perder tempo aprendendo a usar outra interface.	1	7,69%
O preço.	2	15,38%
Outros	0	0,00%
Total	13	

Pergunta 4 (tab. 4).

- A maioria (sete usuários) respondeu que ***dominar melhor o software*** era um dos principais motivos para usá-lo com mais frequência. Os softwares usados por estes usuários são o Encore (3), Finale (2), abcm2ps (1), TuxGuitar (1).
- Quatro usuários justificaram o uso mais frequente pela qualidade do resultado final. Os softwares usados por estes usuários são o abcm2ps (1), o Finale (2), o Lilypond (1) e “Sibelius/Finale” (1).
- Quatro usuários responderam que só tinham acesso a um determinado software. Os softwares usados por estes usuários são o Encore (2) e o Finale (2).

- A rapidez é um dos motivos apontados por três usuários. Os softwares usados por estes usuários são o abcm2ps (1), o Encore (1) e o TuxGuitar (1).
- O preço é um dos motivos apontados por dois usuários. Os softwares usados por estes usuários são o Lilypond (1) e o abcm2ps (1), ambos gratuitos. O usuário do TuxGuitar, também gratuito, não marcou a opção preço.
- A facilidade para conseguir ajuda de amigos em caso de dúvida foi marcada por um usuário (Lilypond).
- Ter sido o primeiro software que usou, e não estar disposto a aprender outra interface, é um dos motivos para um usuário (Encore).
- Os motivos apresentados na opção “outros” são: software livre, pequeno e extensível (abcm2ps) e a possibilidade de criar partituras em qualquer computador, usando um editor de textos, para compilar posteriormente (Lilypond). Observa-se que apesar de cada um destes usuários ter apontado características diferentes, elas se aplicam a ambos os softwares.

Tabela 5 – Respostas para a pergunta 5.

5. Qual a sua relação com a música?	número de respostas	porcentagem
Amador.	9	69,23%
Estudante visando à profissionalização.	1	7,69%
Profissional.	3	23,08%
Total	13	

Pergunta 5 (tab. 5).

- A maioria dos usuários que respondeu a pesquisa é de amadores (9).
- Dos três profissionais, um usa o abcpm2ps, um usa o Finale, e um diz usar “Sibelius/Finale”.
- O estudante com intenção de profissionalização diz usar mais lápis e papel.

Tabela 6 – Respostas para a pergunta 6.

6. Com que frequência você costuma editar partituras?	número de respostas	porcentagem
Raramente.	3	23,08%
De vez em quando.	8	61,54%
Frequentemente.	2	15,38%
Total	13	

Pergunta 6 (tab. 6).

- A maioria (8) diz editar partituras “de vez em quando”.
- Entre os dois usuários que dizem editar partituras frequentemente, há um amador (usuário do Encore) e um profissional (usuário de “Sibelius/Finale”).
- Os três usuários que editam partituras raramente são o estudante com intenção de profissionalização, e dois amadores (um usuário do Encore e um usuário do Finale).

Tabela 7 – Respostas para a pergunta 7.

7. Para que você usa softwares de edição de partituras?	número de respostas	porcentagem
Registrar (para seu uso pessoal) suas próprias composições ou arranjos.	9	69,23%
Divulgação ou comercialização de suas próprias composições e arranjos.	2	15,38%
Registro ou cópia de partituras para um trabalho em grupo (ex. duo, quarteto, coral) que você participa.	5	38,46%
Para trabalhos acadêmicos.	3	23,08%
Como um serviço profissional para terceiros (inclusive editoras).	2	15,38%
Não respondeu.	2	15,38%
Outros	0	0,00%
Total	13	

Pergunta 7 (tab. 7).

- O uso mais frequente (9) é o registro pessoal de suas próprias composições e arranjos.
- Cinco usuários produzem partituras para uso com um grupo (coral, duo, quarteto).
- Três usuários editam partituras para uso em trabalhos acadêmicos.
- Dois usuários editam partituras visando à divulgação ou comercialização de suas próprias composições e arranjos.
- Dois usuários editam partituras como um serviço para terceiros. Estes são o usuário de abcm2ps e o usuário de “Sibelius/Finale” (daí talvez uma grande utilização de Finale mesmo afirmando preferir o Sibelius – o formato pode ser exigência de quem contratou o serviço).
- Dois usuários não responderam.
- Dentre os três profissionais, um marcou todas as opções, um marcou registro pessoal, uso com grupos e trabalhos acadêmicos, e um marcou apenas serviço para terceiros.

Pergunta 8: Você costuma editar partituras predominantemente para algum instrumento ou formação instrumental específica? Qual?

- Como era previsível, em virtude dos usuários convidados serem em sua maioria membros de um fórum de discussão sobre violão, a maioria (10) edita partituras para violão solo.
- Um usuário disse não editar predominantemente para nenhum instrumento ou formação instrumental específica.
- Os outros usos incluem a edição de partituras para conjuntos (duos, quartetos), coral, teclado, piano, violino, guitarra e baixo. O usuário que edita para guitarra e baixo usa o TuxGuitar.

Tabela 8 – Respostas para a pergunta 9.

9. Você tem experiência com computadores?	número de respostas	porcentagem
Muito pouca.	0	0,00%
Costuma usar internet, e alguns aplicativos, mas não sempre.	0	0,00%
Usa o computador todo dia, mas não trabalha com ele.	4	30,77%
Trabalha usando intensivamente o computador.	7	53,85%
Trabalho no desenvolvimento de sistemas de computador.	2	15,38%
Outros	0	0,00%
Total	13	

Pergunta 9 (tab. 8).

- Em geral, todos os usuários que responderam a pesquisa são experientes no uso de computadores.

Tabela 9 – Respostas para a pergunta 10.

10. Como você classificaria estes softwares quanto à facilidade de aprender a usar?	Encore	Finale	Lilypond	Sibelius	Outros
Facílimo (é só sair usando, imediatamente)	4	0	0	0	1
Fácil (dá pra aprender rapidamente, observando e experimentando)	6	0	0	1	0
Médio (dá pra aprender sozinho, lendo manuais, ajuda, etc.).	0	7	3	5	1
Difícil (é preciso treinamento ou ajuda especializada)	0	2	1	0	0
Sem opinião	3	4	9	7	0
Total	10	9	4	6	2

Pergunta 10 (tab. 9).

- Na (tab. 9) até a (tab. 16), os totais são relativos apenas aos usuários que opinaram para cada software.
- O Encore foi considerado de “facílimo” a “fácil”.
- O Sibelius foi considerado de “fácil” a “médio” (predominantemente “médio”).
- O Finale foi considerado de “médio” a “difícil” (predominantemente “médio”).
- O Lilypond foi considerado de “médio” a “difícil” (predominantemente “médio”).
- Dentre os “outros”, um usuário considerou o “TuxGuitar” “facílimo”, e um usuário considerou o abcm2ps “médio”.

Tabela 10 – Respostas para a pergunta 11.

11. Como você classificaria estes softwares quanto à eficiência?	Encore	Finale	Lilypond	Sibelius	Outros
Muito Baixa	2	0	0	0	0
Baixa	1	0	1	1	0
Média	6	2	1	3	0
Alta	0	6	1	3	1
Sem opinião	4	5	10	6	12
Total	9	8	3	7	1

Pergunta 11 (tab. 10).

- A eficiência do Encore é considerada de “muito baixa” a “média”, com maior concentração em “média”.
- A eficiência do Finale é considerada de “média” a “alta”, com maior concentração em “alta”.
- A eficiência do Sibelius é considerada de “baixa” a “alta”, com maior concentração em “média” e “alta”.
- Cada um dos três usuários que avaliou o Lilypond neste quesito respondeu de forma diferente.
- A eficiência do abcm2ps é considerada “alta” pelo seu usuário.

Tabela 11 – Respostas para a pergunta 12.

12. Você acha que na tentativa de facilitar o aprendizado de novos usuários o software acaba prejudicando o desempenho dos usuários experientes de alguma forma?	Encore	Finale	Lilypond	Sibelius	Outros
Sim	2	0	0	0	0
Não (a interface possui atalhos para os usuários experientes)	3	5	2	3	1
Sem opinião	8	8	11	10	12
Total	5	5	2	3	1

Pergunta 12 (tab. 11).

- Muitos usuários não opinaram nesta pergunta. Talvez ela não seja suficientemente clara.
- Dentre os que opinaram, apenas dois usuários acharam que os recursos voltados para usuários novatos podem prejudicar o desempenho de usuários experientes, no caso do Encore.

Tabela 12 – Respostas para a pergunta 13.

13. Se você já sabe usar bem o software mas fica muito tempo sem usá-lo, é fácil de se lembrar?	Encore	Finale	Lilypond	Sibelius	Outros
Sim (dá pra lembrar rapidamente só tentando usar de novo)	9	0	0	0	1
Um pouco (algumas coisas são mais difíceis de lembrar, mas não demora muito pra recordar)	1	5	1	4	1
Não (é preciso aprender a usar tudo de novo)	0	3	2	1	0
Sem opinião	3	5	10	8	11
Total	10	8	3	5	2

Pergunta 13 (tab. 12).

- O Encore é considerado fácil de lembrar por usuários casuais.
- Na percepção dos usuários, o Finale e o Sibelius exigem um pouco mais do usuário casual.
- O Lilypond é considerado mais difícil de lembrar.
- O usuário de abcm2ps considera o software um pouco exigente para o usuário casual, e o usuário de TuxGuitar o considera fácil de lembrar.

Tabela 13 – Respostas para a pergunta 14.

14. Você costuma cometer algum erro ou demorar mais em algumas tarefas porque a interface não é clara ou alguma funcionalidade não se comporta como você acha que deveria?	Encore	Finale	Lilypond	Sibelius	Outros
Sim	7	5	1	3	0
Não	2	2	1	1	2
Sem opinião	4	6	11	9	11
Total	9	7	2	4	2

Pergunta 14 (tab. 13).

- Todos os principais softwares apresentam problemas neste quesito.

Tabela 14 – Respostas para a pergunta 15.

15. Neste caso, o software pelo menos permite identificar e corrigir facilmente estes erros?	Encore	Finale	Lilypond	Sibelius	Outros
Sim	2	0	0	0	1
Não	6	4	1	3	0
Sem opinião	5	9	12	10	12
Total	11	4	1	3	1

Pergunta 15 (tab. 14).

- Todos os principais softwares apresentam problemas neste quesito.
- De acordo com as respostas às perguntas 14 e 15, todos os principais softwares poderiam se beneficiar por melhorias em relação às heurísticas “prevenções de erro” e “boas mensagens de erro”.

Tabela 15 – Respostas para a pergunta 16.

16. É fácil e rápido encontrar ajuda no help e nos manuais quando você precisa?	Encore	Finale	Lilypond	Sibelius	Outros
Sim	3	2	2	1	1
Não	3	4	0	3	0
Sem opinião	7	7	11	9	12
Total	6	6	2	4	1

Pergunta 16 (tab. 15).

- Na opinião da maioria dos usuários, os sistemas de ajuda e documentação do Finale e do Sibelius deixam a desejar.
- Curiosamente, o software mais intuitivo (e para o qual, portanto, a necessidade de ajuda e documentação é menor) dos três com interface gráfica, o Encore, apresenta uma impressão um pouco melhor (embora longe do ideal) neste quesito.
- As documentações dos softwares sem interface gráfica (Lilypond, abcm2ps) são consideradas boas.

Tabela 16 – Respostas para a pergunta 17.

17. Você acha o software agradável?	Encore	Finale	Lilypond	Sibelius	Outros
Sim	6	3	2	4	2
Não	1	4	2	1	0
Sem opinião	6	6	9	8	11
Total	7	7	4	5	2

Pergunta 17 (tab. 16).

- O Encore e o Sibelius são considerados “agradáveis”.
- Menos usuários acharam o Finale agradável do que o contrário.
- Em relação ao Lilypond, houve um empate de opiniões.
- Esta é uma pergunta bastante subjetiva. Houve um grande número de abstenções, e até um comentário em relação à não entender a intenção da questão.

A pergunta 18 foi usada como espaço para indicar o software referido em “outros” nas perguntas 10 a 17, caso necessário.

A pergunta 19 foi “*O que você mais gosta e menos gosta em cada software?*”. As repostas são reproduzidas no anexo III, agrupadas por software. Seguem alguns comentários sobre essas respostas.

- Os comentários sobre o Encore apontam uma facilidade de uso para tarefas mais simples e rápidas. As principais reclamações são sobre a qualidade visual da partitura, os erros de diagramação e as limitações para as tarefas mais avançadas.

- O Finale é apontado como um software completo, flexível e que gera partituras de boa qualidade. Entretanto, é considerado difícil de usar, e com uma interface pouco intuitiva (as funções, mesmo as mais básicas, seriam difíceis de encontrar).

- O Sibelius é considerado um software flexível, que gera partituras de ótima qualidade. Não foram apontados pontos negativos.

- Os pontos positivos do Lilypond comentados foram a portabilidade e o fato de ser um programa livre. O ponto negativo foi a necessidade de aprender uma linguagem, considerada complexa.

A pergunta 20 foi um pedido de "*Comentários? Sugestões*?". As repostas são reproduzidas no apêndice IV. Uma análise desses comentários e sugestões é apresentada a seguir.

- Um usuário afirmou não conseguir ser eficiente em programas com interfaces gráficas, especialmente as que requerem o uso intensivo do mouse. Para ele, os programas com interfaces gráficas são mais úteis como auxiliares para composição e arranjo, ou para usuários casuais. Este usuário também ressaltou, no caso dos programas com interfaces gráficas, a importância do posicionamento dos elementos ser baseado em regras automáticas, de forma a evitar inconsistências que poderiam decorrer do livre posicionamento com o mouse.

Estes comentários são pertinentes, e levantam outras questões. Se os programas interativos são mais indicados para usuários casuais, então o atributo "memorabilidade" é crítico, e softwares deficientes neste aspecto não atendem às necessidades destes usuários. Um programa com um algoritmo de diagramação automático que produza resultados pouco agradáveis ou mesmo defeituosos, irá exigir muitas intervenções manuais, o que, além de exigir mais esforço do usuário para completar a tarefa, exige ainda bastante habilidade ("pontaria") com o mouse, que não é um dispositivo apontador de precisão, e um bom conhecimento de regras de diagramação.

- Este usuário, ainda comentando sobre a eficiência, descreveu uma característica muito interessante do abcm2ps. Este software permite representar um trecho repetido, com pequenas diferenças a cada repetição,

sem necessidade de repetir ou copiar este trecho diversas vezes. Em vez disto, há um mecanismo de reuso da codificação do trecho, com as vantagens normalmente associadas ao reuso de código, como facilidade de manutenção (uma alteração só precisa ser feita em um lugar do código, e não em diversas cópias do mesmo trecho).

- Um usuário reclamou da falta de uma interface mais “auto-explicativa” no Finale.

Segundo Krug (2006), na medida do possível, uma interface deve ser “evidente”. Se não puder ser “evidente”, que ao menos procure ser “auto-explicativa”. O livro de Krug é voltado para páginas web, mas este é um conceito que pode servir como orientação no desenvolvimento de outros tipos de aplicativos.

- Um usuário levantou a questão do software livre. Os três softwares mais usados, citados na pesquisa (Encore, Finale e Sibelius), são softwares proprietários. Como são softwares com muitas aplicações na área de educação, a importância de procurar alternativas livres seria ainda maior.

Os três softwares citados oferecem descontos para uso acadêmico. Além disto, o Finale e o Sibelius possuem versões mais simples, com limitações, vendidas a um valor menor. Embora todos utilizem formatos proprietários de arquivos, também permitem a exportação para outros formatos, alguns deles abertos, como o MusicXML. Entretanto, mesmo os preços das versões acadêmicas ainda são elevados para estudantes, e mesmo para muitos músicos profissionais. As versões limitadas são de pouca valia, por exemplo, para um estudante de composição que precisa escrever para uma orquestra ou mesmo para um grupo um pouco maior. A aplicação destes softwares em educação, na formação de novos artistas, na produção e divulgação de cultura, realmente reforça a importância de se procurar alternativas mais acessíveis. Existem outros aspectos nesta questão do software livre versus software proprietário, além do preço, que ficam além do escopo deste trabalho, que é a usabilidade das interfaces. Fica a sugestão de uma investigação sobre estes aspectos como tema para futuros trabalhos.

III - CONCLUSÃO

A avaliação heurística apontou problemas de usabilidade em potencial em todos os softwares testados. Muitos destes problemas poderiam ser corrigidos em novas versões, caso fossem detectados em uma avaliação que fizesse parte do ciclo de desenvolvimento destes softwares. Muitas vezes, entretanto, os desenvolvedores estão sobrecarregados com a correção de defeitos (bugs), ou priorizam a incorporação de cada vez mais e mais funcionalidades, e relegam a usabilidade a um segundo plano.

O software Finale, entretanto, apresentou um problema de usabilidade em potencial que é estrutural, e que não poderia ser corrigido sem uma mudança radical em sua interface. O Finale é completamente baseado em modos, o que é contrário às recomendações dos especialistas em usabilidade. Além disto, o seu sistema de menus precisaria de uma reorganização profunda para se tornar mais lógico e natural. Isto cria um impasse para os desenvolvedores. Mantendo a interface atual, o Finale vem perdendo cada vez mais mercado, principalmente para o Sibelius, e mesmo para softwares como o Lilypond e o abcm2ps (que atraem os usuários dispostos a abrir mão de uma interface interativa desde que obtenham uma maior eficiência e resultados de alta qualidade). Uma mudança radical, entretanto, teria o efeito de criar dificuldades de uso para os usuários mais antigos, que já estão habituados às características da interface atual, e para os quais ela já se tornou natural. Como a recomendação de usuários experientes e a disponibilidade destes para ajudá-los é um fator importante considerado por usuários novatos para a escolha de um software, uma mudança deste porte poderia surtir o efeito inverso ao desejado, levando a uma perda ainda maior de posição no mercado.

As repostas ao questionário confirmaram as impressões adquiridas sobre os softwares durante a avaliação heurística. Além disto, pontos adicionais muito interessantes foram levantados.

O Encore é simples de usar, sendo adequado para tarefas mais simples, mas peca

em eficiência e na qualidade dos resultados, o que o torna pouco recomendável para tarefas maiores ou mais sérias. Entretanto, mesmo com problemas de qualidade e até defeitos, dentre os participantes do questionário, é o que teve mais adeptos, o que mostra como a facilidade de uso é um fator importante para os usuários na escolha de um software.

A impressão dos participantes sobre o Finale confirma o que foi observado na avaliação heurística: um software com uma interface pouco intuitiva, difícil de aprender e de lembrar, mas completo, flexível e que produz resultados de alta qualidade.

O Sibelius se encontra numa posição curiosa. Os participantes o julgaram de uso mais fácil que o Finale, mas não tanto quanto o Encore. Foi o único dos três para o qual não foram apontados pontos negativos sobre a sua usabilidade. É considerado completo, flexível e capaz de produzir resultados de alta qualidade. Entretanto é utilizado por uma parcela pequena dos participantes.

Os softwares sem interface interativa, como o Lilypond e o abcm2ps, são utilizados por poucos participantes, mas estes afirmam que tais softwares são muito eficientes e produzem resultados com excelente qualidade.

A questão do software livre foi levantada por um participante, e é especialmente importante pela aplicação dos sistemas avaliados em educação, produção e divulgação de cultura. O Lilypond e o abcm2ps são softwares livres, portáveis, e que produzem resultados de qualidade. O TuxGuitar foi apontado como um software livre com uma interface com boa usabilidade, mas é um software voltado para uma aplicação específica (guitarras e instrumentos de corda dedilhada). Um software livre de uso geral, completo, flexível, que produza resultados de qualidade, e possua uma interface eficiente, fácil de aprender e agradável de usar, seria uma alternativa bem vinda aos softwares proprietários avaliados.

ANEXO I - QUESTIONÁRIO

Formulário (questionário on-line).

http://spreadsheets.google.com/viewform?hl=pt_BR&formkey=dDl2UUtWVlZiR2tCNmdmRzFfMzFZaUE6MA

Usabilidade de Softwares de Edição de Partituras

Estou fazendo um trabalho sobre avaliação da usabilidade de interfaces de softwares de editoração de partituras. Se vocês puderem dar a opinião de vocês sobre as interfaces de alguns softwares, me ajudaria muito.

1. Você usa ou já usou algum software de edição de partituras?
(pode marcar mais de uma opção)

- [ ] Não, só lápis e papel.
- [ ] Encore
- [ ] Finale
- [ ] Lilypond
- [ ] Sibelius
- [ ] Outro:

2. Dentre os softwares que você marcou qual é o seu preferido?

3. E qual o que você usa mais?
(Não precisa ser a mesma resposta da pergunta anterior)

4. Quais os principais motivos pelos quais você usa mais este software?

- [ ] A qualidade do resultado final.
- [ ] É mais rápido.
- [ ] É o que eu sei usar melhor.
- [ ] Meu trabalho exige a partitura neste formato.
- [ ] É o único ao que eu tenho acesso.
- [ ] Porque é mais fácil conseguir ajuda de amigos quando tenho alguma dúvida.
- [ ] Foi o primeiro que eu aprendi a usar e não quero experimentar outro para não perder tempo aprendendo a usar outra interface.

- [ ] O preço.
- [ ] Outro:

5. Qual a sua relação com a música?
(escolha uma opção)

- [ ] Amador.
- [ ] Estudante visando a profissionalização.
- [ ] Profissional.
- [ ] Outro:

6. Com que frequência você costuma editar partituras?
(escolha uma opção)

- [ ] Raramente.
- [ ] De vez em quando.
- [ ] Frequentemente.

7. Para que você usa softwares de edição de partituras?
(pode marcar mais de uma opção)

- [ ] Registrar (para seu uso pessoal) suas próprias composições ou arranjos.
- [ ] Divulgação ou comercialização de suas próprias composições e arranjos.
- [ ] Registro ou cópia de partituras para um trabalho em grupo (ex. duo, quarteto, coral) que você participa.
- [ ] Para trabalhos acadêmicos.
- [ ] Como um serviço profissional para terceiros (inclusive editoras).
- [ ] Outro:

8. Você costuma editar partituras predominantemente para algum instrumento ou formação instrumental específica? Qual?
(violão solo, coral, piano, etc.)

9. Você tem experiência com computadores?

- [ ] Muito pouca.
- [ ] Costuma usar internet, e alguns aplicativos, mas não sempre.
- [ ] Usa o computador todo dia, mas não trabalha com ele.
- [ ] Trabalha usando intensivamente o computador.
- [ ] Trabalho no desenvolvimento de sistemas de computador.
- [ ] Outro:

10. Como você classificaria estes softwares quanto à facilidade de aprender a usar?
(não precisa classificar todos - só aqueles que você já experimentou)

	Fácilimo *(é só sair usando, imediatamente)*	*Fácil* *(dá pra aprender rapidamente, observando e experimentando)*	*Médio* *(dá pra aprender sozinho, lendo manuais, ajuda, etc.)*	*Difícil* *(é preciso treinamento ou ajuda especializada)*	*Sem opinião*
Encore	☐	☐	☐	☐	☐
Finale	☐	☐	☐	☐	☐
Lilypond	☐	☐	☐	☐	☐
Sibelius	☐	☐	☐	☐	☐
"outros" (indique qual ao final do formulário)	☐	☐	☐	☐	☐

11. Como você classificaria estes softwares quanto à eficiência? Um usuário experiente consegue resultados de alta qualidade em pouco tempo?
(não precisa classificar todos - só aqueles que você conhecer melhor)

	Muito Baixa	*Baixa*	*Média*	*Alta*	*Sem opinião*
Encore	☐	☐	☐	☐	☐
Finale	☐	☐	☐	☐	☐
Lilypond	☐	☐	☐	☐	☐
Sibelius	☐	☐	☐	☐	☐
"outros" (indique qual ao final do formulário)	☐	☐	☐	☐	☐

12. Você acha que na tentativa de facilitar o aprendizado de novos usuários o software acaba prejudicando o desempenho dos usuários experientes de alguma forma?

	Sim)	*Não* *(a interface possui atalhos para os usuários experientes)*	*Sem opinião*
Encore	☐	☐	☐
Finale	☐	☐	☐
Lilypond	☐	☐	☐
Sibelius	☐	☐	☐
"outros" (indique qual ao final do formulário)	☐	☐	☐

13. Se você já sabe usar bem o software mas fica muito tempo sem usá-lo, é fácil de se lembrar?

	Sim *(dá pra lembrar rapidamente só tentando usar de novo)*	*Um pouco* *(algumas coisas são mais difíceis de lembrar, mas não demora muito pra recordar)*	*Não* *(é preciso aprender a usar tudo de novo)*	*Sem opinião*
Encore	☐	☐	☐	☐
Finale	☐	☐	☐	☐
Lilypond	☐	☐	☐	☐
Sibelius	☐	☐	☐	☐
"outros" (indique qual ao final do formulário)	☐	☐	☐	☐

14. Você costuma cometer algum erro ou demorar mais em algumas tarefas porque a interface não é clara ou alguma funcionalidade não se comporta como você acha que deveria?

	Sim)	*Não*	*Sem opinião*
Encore	☐	☐	☐
Finale	☐	☐	☐
Lilypond	☐	☐	☐
Sibelius	☐	☐	☐
"outros" (indique qual ao final do formulário)	☐	☐	☐

15. Neste caso, o software pelo menos permite identificar e corrigir facilmente estes erros?

	Sim)	*Não*	*Sem opinião*
Encore	☐	☐	☐
Finale	☐	☐	☐
Lilypond	☐	☐	☐
Sibelius	☐	☐	☐
"outros" (indique qual ao final do formulário)	☐	☐	☐

16. É fácil e rápido encontrar ajuda no help e nos manuais quando você precisa? E a informação é útil?

	Sim)	*Não*	*Sem opinião*
Encore	☐	☐	☐
Finale	☐	☐	☐
Lilypond	☐	☐	☐
Sibelius	☐	☐	☐
"outros" (indique qual ao final do formulário)	☐	☐	☐

17. Você acha o software agradável?
(O uso do software é uma experiência agradável ou um incômodo?)

	Sim)	*Não*	*Sem opinião*
Encore	☐	☐	☐
Finale	☐	☐	☐
Lilypond	☐	☐	☐
Sibelius	☐	☐	☐
"outros" (indique qual ao final do formulário)	☐	☐	☐

18. Nome do software (outros):
(Complementando as perguntas acima)

19. O que você mais gosta e menos gosta em cada software?
Não é preciso comentar todos.

20. Comentários? Sugestões? Obrigado por colaborar com esta pesquisa ☺!

ANEXO II – GRÁFICOS (REPOSTAS DO QUESTIONÁRIO)

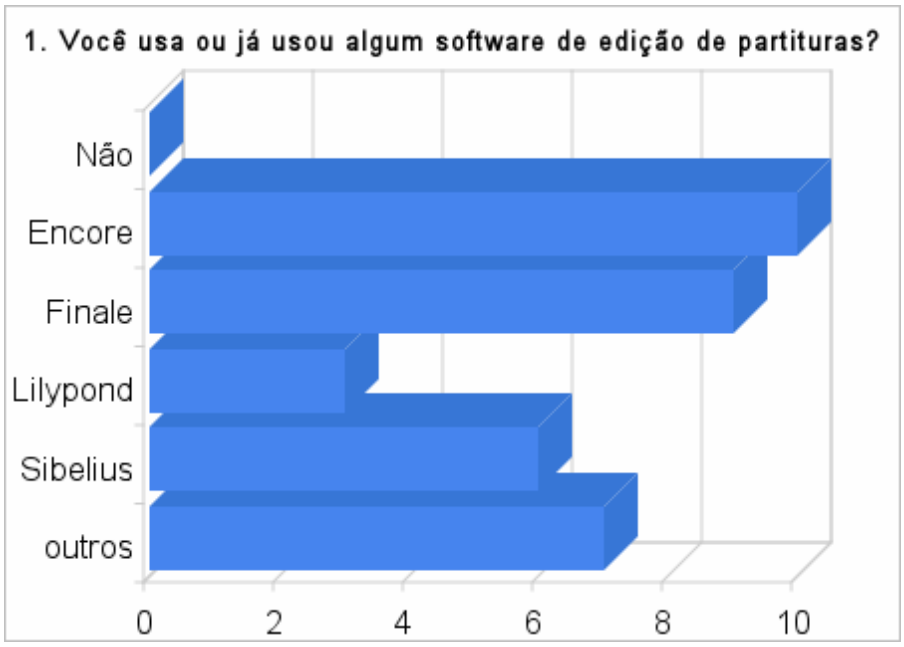

Figura 17 - Respostas para a pergunta 1.

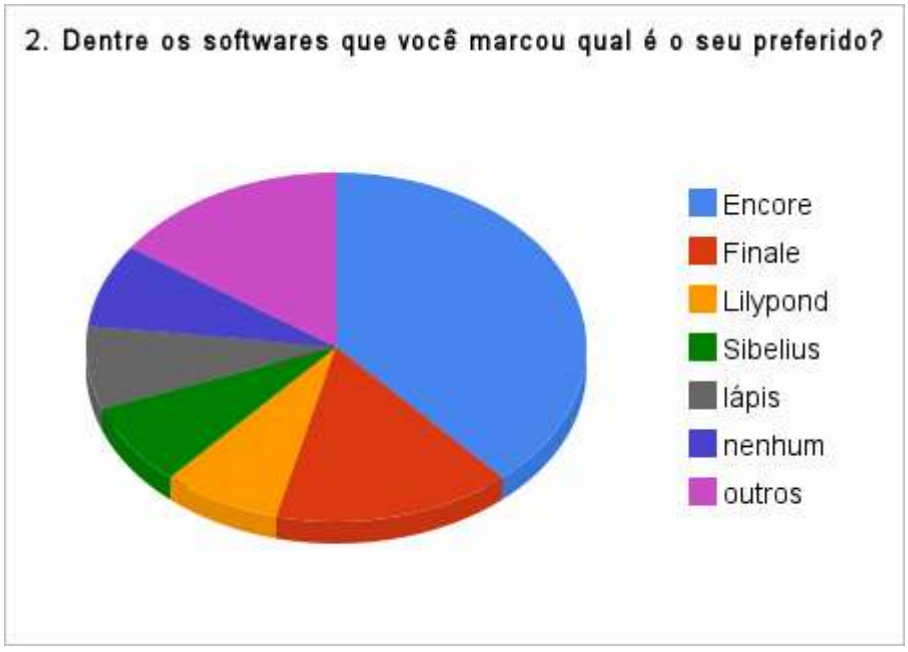

Figura 18 - Respostas para a pergunta 2.

Figura 19 - Respostas para a pergunta 3.

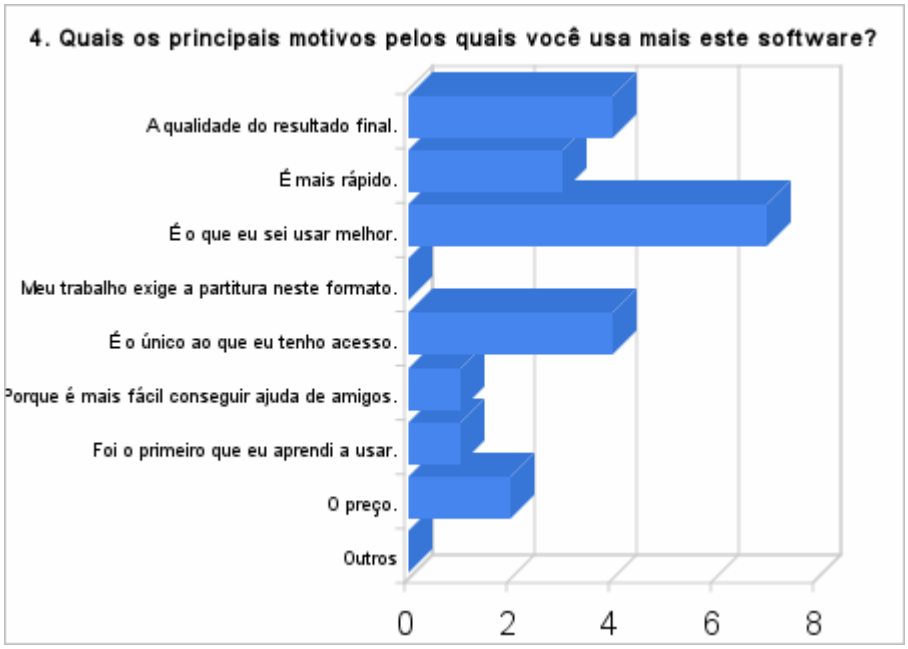
4. Quais os principais motivos pelos quais você usa mais este software?
A qualidade do resultado final.
É mais rápido.
É o que eu sei usar melhor.
Meu trabalho exige a partitura neste formato.
É o único ao que eu tenho acesso.
Porque é mais fácil conseguir ajuda de amigos.
Foi o primeiro que eu aprendi a usar.
O preço.
Outros
0
2
4
6
8

Figura 20 - Respostas para a pergunta 4.

Figura 21 - Respostas para a pergunta 5.

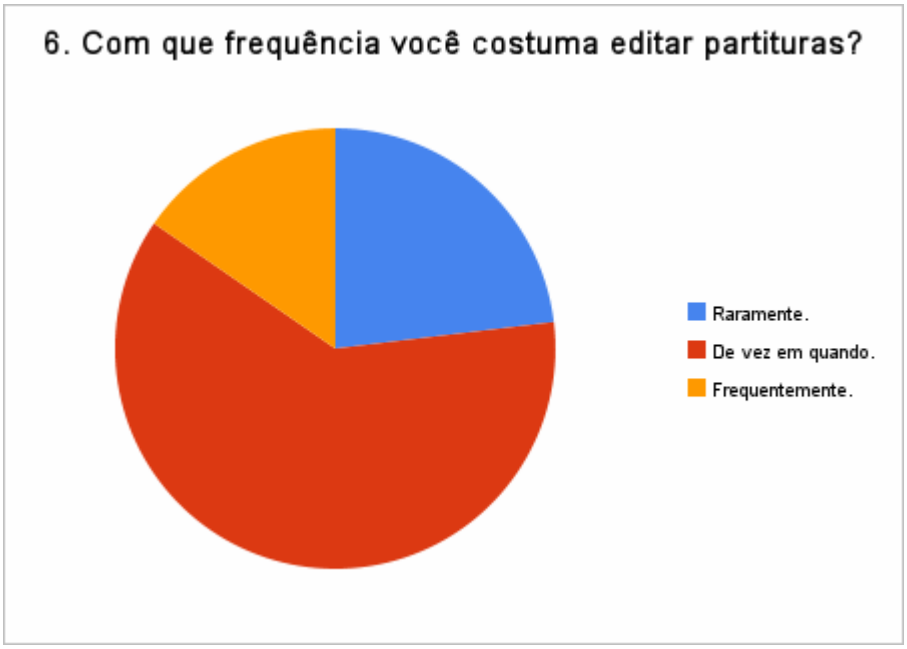

Figura 22 - Respostas para a pergunta 6.

Figura 23 - Respostas para a pergunta 7.

Figura 24 - Respostas para a pergunta 9.

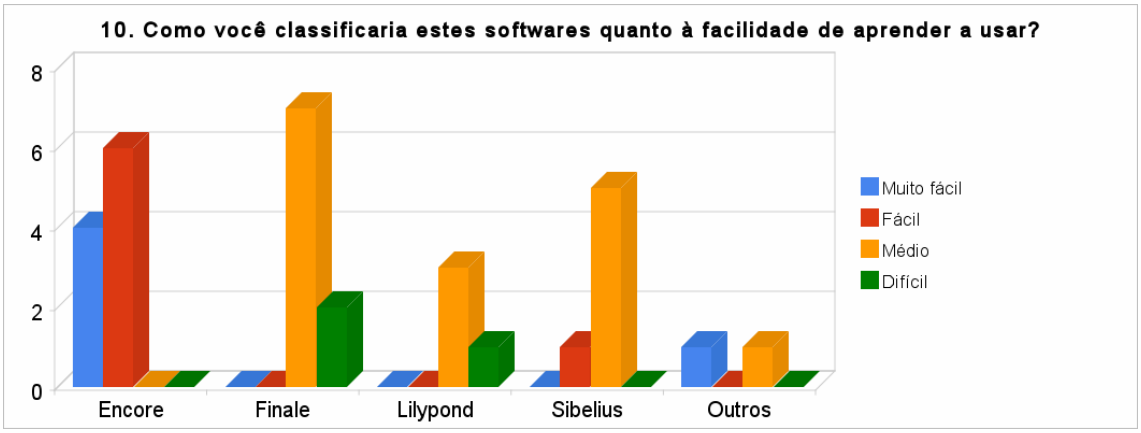

Figura 25 - Respostas para a pergunta 10.

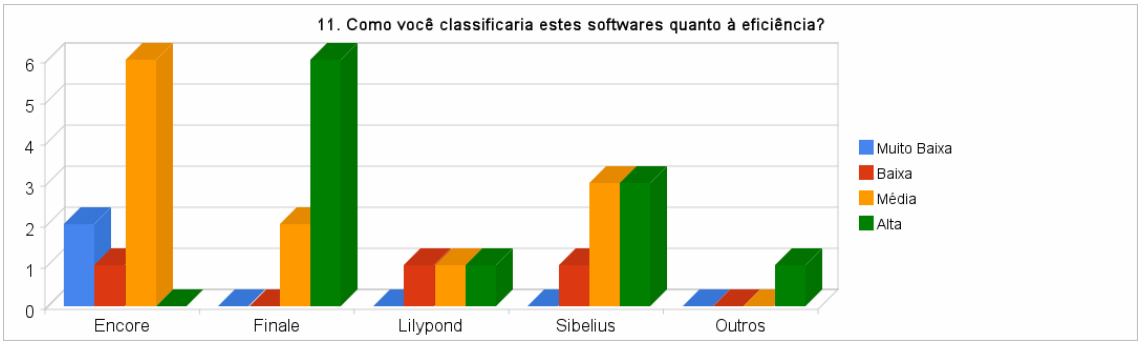

Figura 26 - Respostas para a pergunta 11.

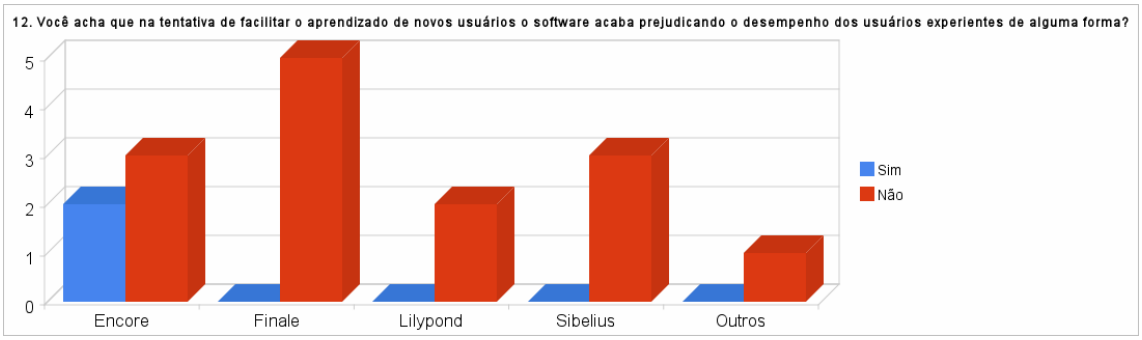

Figura 27 - Respostas para a pergunta 12.

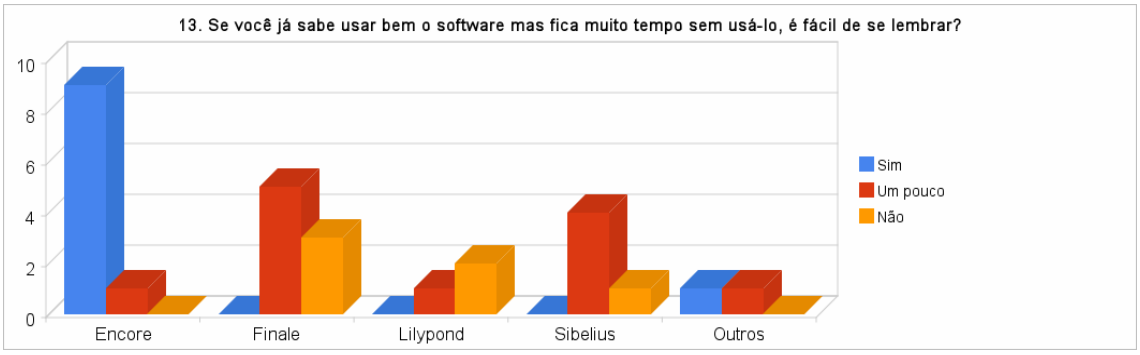

Figura 28 - Respostas para a pergunta 13.

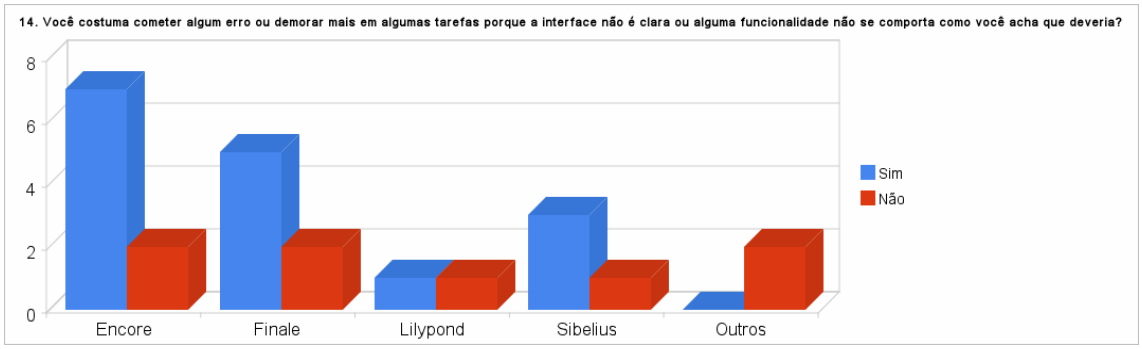

Figura 29 - Respostas para a pergunta 14.

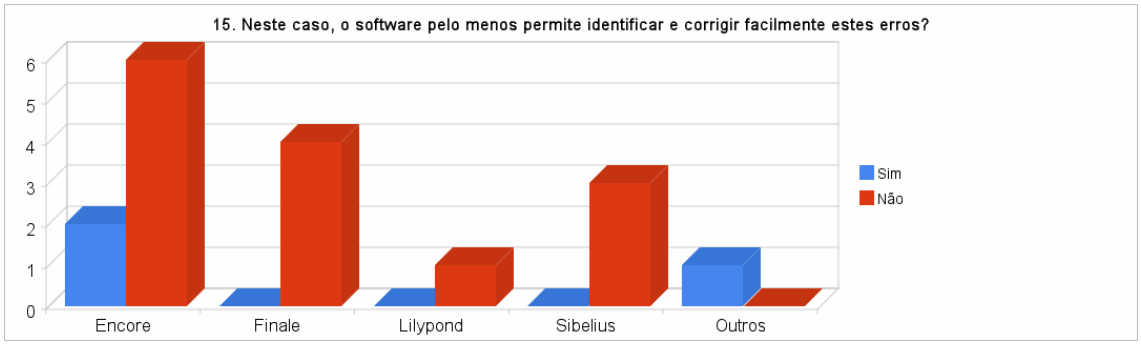

Figura 30 - Respostas para a pergunta 15.

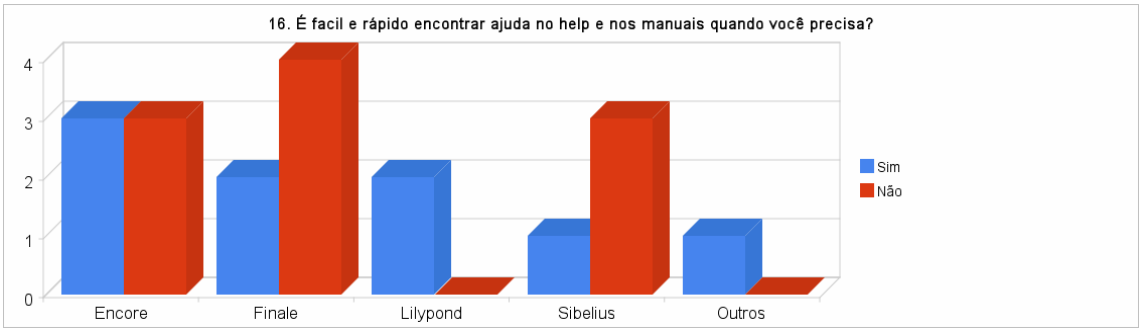

Figura 31 - Respostas para a pergunta 16.

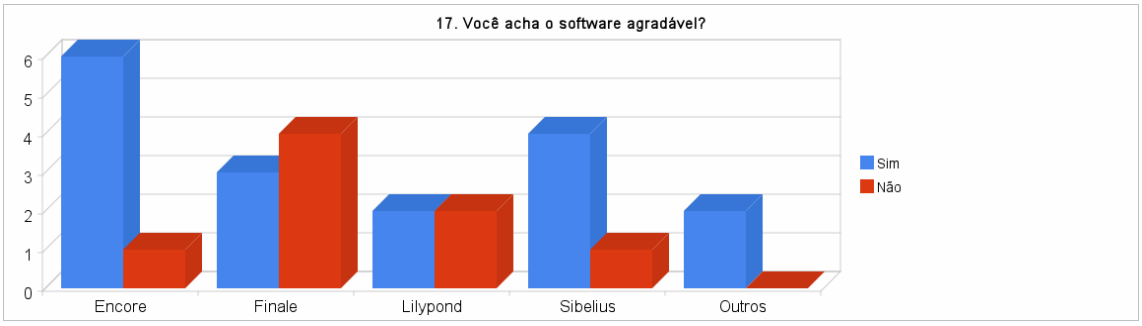

Figura 32 - Respostas para a pergunta 17.

ANEXO III – PERGUNTA 19 (QUESTIONÁRIO)

PONTOS POSITIVOS

Encore

“O que mais gosto é que praticamente todas as tarefas podem ser realizadas utilizando o mouse e elementos disponíveis na tela.”

“Facilidade de aprendizado e uso. Ótimo para tarefas básicas como linha melódica/letra/cifra.”

“Gosto da facilidade do uso.”

“Acho relativamente fácil e rápido para fazer coisas básicas.”

“Bem prático e objetivo. A última versão do Encore resolveu o problema de travar as tarefas[1]. Apesar de limitado ele atende as minhas necessidades.”

“O Encore é de fácil acesso.”

Finale

“Solidez de arquivo. Altíssima flexibilidade e controle de ferramentas. Cria arquivos de exportação de boa qualidade.”

“O Finale é bastante completo.”

[1] A versão utilizada na avaliação heurística era a última disponível para PC no site da GVOX e mesmo assim apresentou problemas de travamento. Talvez este usuário use uma plataforma diferente (Apple) ou tenha havido um problema de incompatibilidade com algum componente do PC utilizado nos testes.

Lilypond

“Gosto muito da portabilidade.”

Sibelius

“O que mais gosto é a janela representando o teclado numérico, ajuda a lembrar os atalhos.”

“Solidez de arquivo. Alta flexibilidade e controle de ferramentas. Cria arquivos de exportação específicos para vários programas de ótima qualidade. Fonte musical (Opus) de excepcional legibilidade mesmo em tamanhos reduzidos.”

TuxGuitar

“É extremamente fácil de utilizar, e apesar de ser voltado para instrumentos de corda, permite edição diretamente na pauta, portanto pode ser utilizado para escrita de qualquer instrumento. Apresenta um grande banco de acordes para serem utilizados, é uma grande alternativa ao software proprietário Guitar Pro.”

“A principal vantagem desses dois softwares[2] é o fato deles serem softwares livres, portanto estão acessíveis a qualquer pessoa, oferecendo a liberdade que somente um software livre pode oferecer. E cabe dizer que não utilizo softwares proprietários por questões éticas, políticas e filosóficas.”

PONTOS NEGATIVOS

Encore

“O que menos gosto é que não dá pra editar usando só atalhos de teclado, tem que

[2] Lilypond e TuxGuitar.

ficar alternando entre teclado e mouse.”

“Fraquíssima ferramenta de ornamentos. Ligadura entre notas de diferentes alturas de má qualidade visual necessitando sempre correção manual. (Obs.: última versão utilizada deste software foi 4.5.5)”

“Não gosto da dificuldade de representar apogiaturas e mordentes.”

“... mas é muito incompleto e sensível a mudanças de contexto.”

“... mas é limitado em recursos.”

Finale

“O que menos gosto é que os modos de entrada usam atalhos diferentes pras mesmas coisas, fica difícil memorizar.”

“Curva de aprendizado mais longa entre os 3 softwares (se é que isso pode ser considerado um ponto negativo).”

“... mas não acho fácil de encontrar as funções, às vezes as mais básicas.”

Lilypond

“Sinto falta de uma escrita um pouco mais simplificada.”

“O Lilypond utiliza-se de uma linguagem especifica de programação para a criação das partituras, portanto além de saber ler e escrever música bem, o usuário precisa dominar a tal linguagem, o que torna o processo de criação demorado e cansativo.”

ANEXO IV – PERGUNTA 20 (QUESTIONÁRIO)

“Nos casos do LilyPond e do abcm2ps, que não têm interface interativa, considerei como "interface" a combinação da linguagem + recursos do programa. Em geral, não consigo ser eficiente em programas com interface gráfica, que acho muito desagradáveis de usar (especialmente quando requerem uso do mouse). Prefiro interfaces de texto e atalhos de teclado. No caso de programas que gerenciam elementos gráficos, como é o caso dos editores de partituras, acho essencial que o controle do posicionamento não dependa de pontaria com o mouse, mas seja baseado em regras de posicionamento e alinhamento e parâmetros numéricos. Assim as coisas não ficam tortas como ocorre em muitos programas quando se inserem elementos com o mouse. Um fator que interfere na eficiência é a possibilidade de automatizar tarefas repetitivas. Fiz hoje no LilyPond, por exemplo, uma partitura com forma |:A:||:B:|A|:C:|A (com casas 1 e 2 nos ritornellos), sendo que cada repetição de A é ligeiramente diferente, sem reescrever a seção. Apenas marquei cada uma das pequenas diferenças, então selecionando condicionalmente a versão apropriada ao reutilizar a seção A (armazenada como uma variável). Com outros programas, teria que fazer cópias da seção A, o que traz a desvantagem de, na revisão, ter que corrigir o mesmo erro em vários lugares. Os programas visuais podem ser bons como auxiliares para composição e arranjo (em que pese também sua influência sobre o resultado musical, que tende a ficar moldado ao que é mais fácil de fazer no programa), ou para quem só uma vez ou outra por milênio edita uma partitura, pois em geral possibilitam fazer partituras simples sem muita consulta à documentação.”

“Boa pesquisa!”

“O formulário está bacana, parabéns. Como sugestão para as próximas... evite opções como a da pergunta 6. Coisas como muito, pouco, provável, raramente... são muito subjetivas. O que é pouco pra você será pouco pra mim? :) Mas, como disse, o formulário está bem legal e a pesquisa é muito interessante. Toda sorte!”

“O finale deveria investir numa interface mais "auto-explicativa", ter que ficar buscando as funções de maneira demorada acaba limitando o uso para os usuários menos experientes e diminuindo a qualidade das partituras.”

“Sugiro que se possível aborde, cite a questão, a causa do software livre em seu trabalho. Um fato que ocorre com muitos usuários "religiosos" de softwares livres, de sistemas operacionais GNU/Linux, inclusive comigo é utilizar o critério da liberdade (é um software livre ou não?) para dizer se um software é bom ou não, em detrimento da facilidade, da qualidade da interface e sua usabilidade. O fato é que a maioria dos softwares que você citou são proprietários e bem caros, portanto, não acessíveis para uma maioria, que se quiserem utilizá-los, apelarão para o que as corporações chamam de "pirataria". Com isso o software livre cumpre uma função social e política importante, e como estes softwares de edição de partituras são também softwares educacionais, fica ainda mais claro a importância de termos ferramentas livres para a educação de crianças, adolescentes e adultos que não podem pagar, ainda que a questão do valor não seja a questão principal do software livre, e sim a liberdade, que é algo que não tem preço. Abraço.”

REFERÊNCIAS BIBLIOGRÁFICAS

NIELSEN, J. **Usability Engineering**. Boston: Academic Press, 1993.

LASSFOLK, K. **Music Notation as Objects**: An Object-Oriented Analysis of the Common Western Music Notation System. Helsinki: 2004. Disponível em <http://ethesis.helsinki.fi/julkaisut/hum/taite/vk/lassfolk/musicnot.pdf>. Acesso em 5 de outubro 2009.

PLAY IT AGAIN! PIANO STUDIO. **History of Music Notation**. Disponível em <http://playitagainpianostudio.com/Documents/History%20of%20Music%20Notation.pdf>. Acessado em 5 de outubro 2009.

BLOOD, B. **Music Theory Online**: Notes & Rests (lesson 2). Disponível em <http://www.dolmetsch.com/musictheory2.htm>. Acesso em 5 de outubro 2009.

REYNA, R. **The History of Music Printing Museum**. Disponível em <http://www.musicprintinghistory.org/index.php>. Acesso em 6 de outubro 2009.

THEORETICALLY CORRECT. **Musical Notation Sotware.** Disponível em <http://www.theoreticallycorrect.com/Encore-5-review/index.html>. Acesso em 1º de março de 2010.

BUSINESS WIRE. **Net4Music Inc. to Change Name to MakeMusic! Inc.** Business Wire, Minneapolis, 13 de março de 2002. Disponível em <http://www.thefreelibrary.com/Net4Music Inc. to Change Name to MakeMusic! Inc.-a083728565>. Acessado em 1º de março de 2010.

BIELAWA, H. **Sibelius 7 Music Notation Software.** Computer Music Journal, Cambridge, Vol. 21, No. 2 (verão de 1997), p. 99-105.

CULLEN, D. **Avid buys Sibelius for a song.** The Register, Londres, 3 de agosto 2006. Disponível em <http://www.theregister.co.uk/2006/08/03/avid_buys_sibelius/>. Acessado em 1º de março de 2010.

BROWN, J. R. **The world market leader in music notation software Sibelius 5**. (Publicada originalmente na Classical Music Magazine, 21 julho 2007). Disponível em <http://www.john-robert-brown.com/sibelius.htm>. Acesso em 10 de outubro 2009.

MOODS PROJECT. **Music Notation**. Disponível em <http://dsii.dsi.unifi.it/~moods/moods/mus_not.htm>. Acesso em 13 de outubro 2009.

MED, B. **Teoria da Música**. 3. ed. Brasília: Musimed, 1980.

KRUG, S. **Don't Make Me Think!** A common sense approach to web usability. 2. ed. Berkeley: New Riders Publishing, 2006.

Software para edição de partituras. Violão.org. Mensagens disponíveis em <http://www.violao.org/index.php?showtopic=7557&hl=>. Acesso em 29 de novembro 2009.

LILYPOND. **Obsessed With Putting Ink On Paper**. Disponível em <http://lilypond.org/about/automated-engraving/big-page>. Acesso em 13 de outubro 2009.

VERDI, G. **La Traviata - N12 Coro di mattadori spagnuoli**. (Transcrição de Toledo, N. Editada usando LilyPond por Corella, A.) Mutopia Project. Disponível em <http://www.mutopiaproject.org/cgibin/make-table.cgi?Composer=VerdiG>. Acesso em 15 de outubro 2009.

LABIUTIL - LABORATÓRIO DE UTILIZABILIDADE. **Critérios Ergonômicos para Avaliação de Interfaces Homem – Computador por Bastien & Scapin, 1993**. Universidade Federal de Santa Catarina – UFSC.
Disponível em <http://www.labiutil.inf.ufsc.br/CriteriosErgonomicos/Abertura.html>.

Acesso em 3 de novembro 2009.

ERGOLIST. **ErgoList**. Disponível em <http://www.labiutil.inf.ufsc.br/ergolist/>. Acesso em 3 de novembro de 2009.

SPOLSKY, J. **User interface design for programmers**. Berkeley: Apress, 2001.

Hotchkiss, R., Ingalls, M. **Encore User Manual.** Version 5. Manual Edition: 2009.1. East Brunswick: Gvox, 2009.

Finale 2010 User Manual. Eden Prairie: Make Music, 2009.

Clarke, T., Spreadbury, D. **Sibelius 6 Handbook**. Edition 6 April 2009. Londres: Avid, 2009.

Encore 5.0 Windows Demo. [Software] Disponível em <http://www.gvox.com/downloadsForm.php?requested=e50w_demo.exe>. Acesso em 5 de outubro de 2009.

Sibelius 6 Demo. [Software] Disponível em <http://www.sibelius.com/cgi-bin/download/get.pl?com=sh&prod=sibdemo>. Acesso em 5 de outubro de 2009.

Finale 2010 Demo. [Software] Disponível em <http://www.finalemusic.com/store/download.aspx?id=427>. Acesso em 5 de outubro de 2009.